BarCode 2000
Version 7.5

User's Manual

©1996-2007 by Unibar™, Inc.

All Rights Reserved

Unibar™ and BARCODE 2000 are registered trademarks of Unibar™, Inc.

All brand and product names used in this manual are trademarks or registered trademarks of their respective companies.

Rochester Hills, Michigan 48309-3103

(248) 299-5050 (Voice)

(248) 299-5050 ext. 290 (Sales)

(248) 299-5050 ext. 291 (Technical Support)

(248) 299-5052 (Fax)

Email: unibar@unibar.com

World Wide Web: www.unibar.com

Table of Contents

Table of Contents 3

Installation 6

Unibar™ Base Directory 6

Setting the Environment String 6
UNIX / Linux 6
Windows Systems 7

License Key File 7

A Brief Overview 8
Quick Start 8
Moving on: 8

Getting Started 9
Create Label Templates: 9
Use the Label Designer (LDS) to design your label templates. 9

Chapter 1 BARCODE 2000 Components: 10

The Primary Menu Interface: 10

The About Dialog 10
About Tab 11
Detail Tab 11
Registration Tab 12
License Tab 13

The Online Help dialog 14

BARCODE 2000 Suite Components: 15

Introduction to Label Data Dictionary (LDD) 17
Purpose 17
LDD Menu Bar 18
LDD Toolbar 19
LDD Field Types 20
Constant 20
Variable 20
Concatenated 21
Counter 22
Time 24
Alias 25
TXDATA 25
File – Graphic Image storage 27
Substring 28
Alternate 29

Introduction to Label Design System (LDS) 30
Purpose 30
Menu Bar 30

LDS Toolbar 32
LDS Fields 34
Explanation of Label Object Behavior 34
Object Selection 34
Object Movement 34
Object Editing 34
Object Deletion / De-selection 34
Descriptions of Label Objects 35
Text-Data Field 35
Barcode Field 38
Vertical/Horizontal Line Field 41
Rectangle Field 42
Graphic Field 43
RFID Field 44
BARCODE 2000 Designer Dialogs by Appearance or Relevance 47
Label Properties 47
Page Setup Dialog 48
Print Preview Dialog 49
Printer Destination Dialog 50
Add Print Destination Dialog 50
Print Mapping Dialog 51
Label Data Defaults Dialog 53
Add LDD Entry Dialog 54
Create LDD Dialog 55
Introduction to Interactive Data Entry (IDE) 55
Purpose 55
Menu Bar 55
Data Entry 56
Transaction Data Record 57
Print Job 57
Introduction to Configuration Tool 59
Purpose 59
Menu Bar 59
Print Destination Table – Add/Edit page 60
Default Destination 61
Formatter Options 61
Chapter 2 - Label Formatter - Direct Use 63
Example of using the Formatter from a command line 63
Example of using the Formatter by piping the data (In C) 63
Example of using the Formatter by passing a file (In Java) 64
Steps in Creating a Job with Formatter 65
Parameter Listing 65
Copies 66
Data Record Format 66

Delimiter 66
End 66
File Input 67
File Type 67
Job Field 67
Job File 67
Label Name 67
New Page 68
New Row 68
Print Destination 68
Chapter 3 – Specialty Barcode Symbologies 69
Serial Shipping Container Code (SSCC) 69
2D – Two Dimensional Barcodes (PDF417) 71
Appendix A – Printer Support 74
Appendix B – Symbologies 78
Appendix C – Supported File Types 79
ASCII - Fixed Field 79
ASCII - Delimited 79
Appendix D – BARCODE 2000 File Type Identification 80
Appendix E – Creating an ASCII file with Access/Excel 81
ACCESS 81
EXCEL 81
Appendix F – Printing “label formats” to non-printers 82
JPEG, PNG & HTML 82
Default Margins 82
Printing Multiple Labels 82
Appendix G – Configuration File & Print Destination Table 84
General Format 84
Printer Destination Table 84
An Example of the Printer Destination Table: 85
Logical 86
Physical 86
Type 86
Default Printer 86
File Type 86
Defining a Delimiter 86
Capacity 87
Max Labels 87
Max Record (Length) 87

Installation

For installation off the internet go to www.unibar.com for download instructions.

Follow the Quick Setup instructions that will guide you through setting up a Unibar™ base directory and a Printer Destination Table before you print your test label. Then you will be ready to start using the BARCODE 2000 programs.

Unibar™ Base Directory

BARCODE 2000 uses a base directory for storing label templates, configuration files, programs, etc.

The default base directory is "/unibar". The user can change to any desired directory, but if the directory is changed, the BARCODE 2000 programs must be informed. Setting the environment string will identify the desired directory.

The environment string should be set before the BARCODE 2000 programs are run. The commands below can be inserted in a login script so they are automatically executed whenever the user logs on. If the BARCODE 2000 programs cannot properly locate the base directory, the following error message will be displayed:

"Error – can't open <base directory>/BC2000.cfg."

Refer to the "BARCODE 2000 Configuration File" section for details on setting up your Printer Destination Table.

Setting the Environment String

UNIX / Linux

```
$ UNIBAR=/usr/unibar
$ export UNIBAR
```

These commands can be included in start up scripts such as .profile or .cshrc of the user who will spawn the formatter process.

** NOTE - 'C: \UNIBAR' is the default directory that BARCODE 2000 installs in on Windows systems. If you changed the directory, you must reflect that change in the UNIBAR™ system variable. **

Windows Systems

1) Go to Control Panel>System
2) Select the 'Environment' tab. (Under ‘Advanced’ tab for XP)
3) Select 'System Variable' by hitting ALT-V
4) Select the 'Variable' text box
5) Type: UNIBAR
6) Select the 'Value' text box
7) Type: C:\UNIBAR (or the path where Barcode 2000 was installed)
8) Select the Set button
9) Select the OK button

License Key File

BARCODE 2000 reads a license key file to determine the features it should enable. The file is named “unibar.key”. The program looks for the key file in the directory that is specified for the Unibar™ base directory.

A Brief Overview

Unibar's™ BARCODE 2000 is a complete bar code label printing system for multi-platforms, including Windows, Linux, and UNIX. BARCODE 2000 includes a Label Designer (LDS), Interactive Data Entry (IDE), Label Data Definition (LDD) editor, and a Label Formatter (print module). They provide the base functions that most users need to print on-demand or batch labels. The Label Designer is used to create the label templates. The Interactive Data Entry is used to input data into a label template for test printing. The Label Data Definition editor is an easy way to setup and define variables used in a label template. The Label Formatter provides data mapping and printing services.

Quick Start

Quick Start assumes you have successfully installed BARCODE 2000. Follow these instructions and you should have a designed and printed label in a few minutes.

- Start BARCODE 2000 and from LDS select File, New Label
- In the Label Properties Dialog Box, type FirstLabel (or some other name) and click or tab through the LDD Name entry (this will create a new LDD with the name FirstLabel). LDDs are very important but you can learn about them later.
- The default printer type is laser (PCL); if you want a different type (Zebra, SATO, etc.) go to the Printer Designation portion of the Label Properties Dialog and select printer model.
- Click on the Insert Object Barcode from the dropdown or Tool Bar.
- Click on the label canvas where you want the barcode to appear. You have a barcode.
- Add a couple text/data fields and a line and box. To modify field, just double click to bring up field properties box and change attributes. Add a graphic if you like.
- To print the label, go to Print Destination if your printer port is lpt1, lpt2, com1, or for Linux/Unix is the default printer (*lp*) you are ready to print your label! **Note**: If your printer is networked or some other port address you will need to go to the Configuration (Create Printer Destination on Main Menu.)
- Select Print and your label should be printed!

Moving on:

To add complicated fields such as concatenations or counters and to print production labels from your application you will need to learn Label Data Dictionary and Label Formatter features as well as additional Label Designer functions. Please take some time to browse the BARCODE 2000 Manual and practice designing and printing more complicated labels. For an example see the sample labels and master.ldd that should be included with your install.

Getting Started

To Begin

Determine which items on your label will be constant (i.e. your company's return address) and which are variable (perhaps a product #, P.O. #, or customer/vendor information). "Text" is a fixed part of the label template and is not considered data. Data and bar code fields are constant or variable (or other LDD supported versions) data defined in the dictionary. Once assigned, the field types can't be changed, the field must be deleted and a new one created.

Specify the source of the data to use. If you are selecting certain fields from a database, get a list of the fields and their maximum sizes. Also be sure that you know the format of the source file (ASCII: delimited or fixed length).

Know which printer type(s) will be used for the job. Changing the printer after the fact could cause you to have to rethink the whole label because not all printers have the same capabilities.

Know the size of the label you will be using.

Know any industry standards for that label regarding placement of certain information. (Some standard labels have been included for your reference. You also have the option of changing these to include your own information, but maintaining the same basic structure for compliance.)

Create Label Templates:

Normally this data is in a data file or is in the program that will "print" the data for the label. The data strings (records) that BARCODE 2000 processes, which create the labels must be ASCII fixed or delimited.

Use the Label Designer (LDS) to design your label templates.

Using the information collected as referred to above; you can now create your label templates. Although this is a separate process from printing production labels the data source definitions are critical for printing correct labels.

For each field you add to a label, you select the attributes such as Symbology, density, font, height, rotation, etc. For fields that take variable data, you define the data in the LDD.

Note *- The Label Designer has default values to test print barcodes.*

Constant label text is text data and is defined in the label and has no other source.

Chapter 1 BARCODE 2000 Components:

This chapter will introduce you to each of BARCODE 2000 components. BARCODE 2000 is accessed via a menu interface. First we will examine this menu interface, and then each of the four suite components will be examined.

The Primary Menu Interface:

The menu interface gives the user access to the four applications in the BARCODE 2000 suite. These applications will be described in the next section.

When one starts the program, they should see the following "menu" or application control:

The primary function of the menu is to give the user access to the BARCODE 2000 applications; the "menu" offers the user the ability to close the application from the File → Exit menu [or the Ctrl-X key combination.]

Additionally, the user may access the new *About Dialog* [Help → About] which provides important system information for the user. The About Dialog provides the user access to information regarding their installation, their Java environment and virtual machine, their registration data and their licensing information.

The About Dialog

The data provided here is designed to help the user answer questions without leaving the application to allow Unibar Customer service technicians to better help you.

About Tab

The first tab, “About,” provides access to telephone numbers to contact customer service, etc. and the company web site to get further information regarding product offerings.

Figure A-1: Provides access to telephone and web info.

Detail Tab

The second tab, “Detail,” provides access to system information. During a call to Unibar Customer Service, the user may be asked for information about the system on which BARCODE 2000 is installed. Such information as operation system, java runtime environment (“JRE”), the installation directory, etc is helpful to solve many user scenarios.

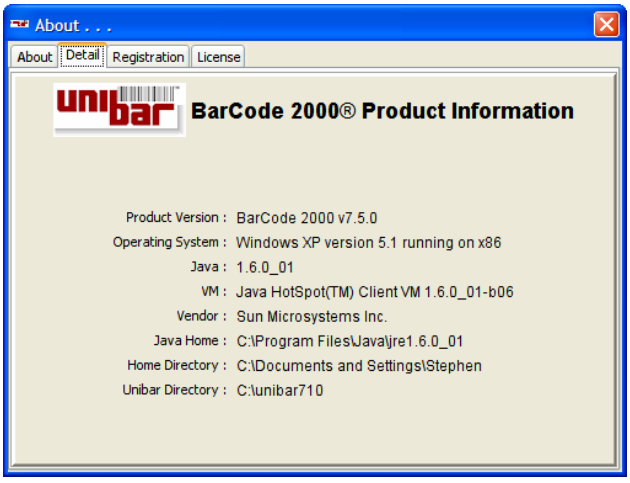


Figure A-2: Provides product and system information.

Registration Tab

The third tab, “Registration,” allows the user to make the command line call to the formatter using the “-ir” switch directly from the User interface, thus eliminating the need to open a separate console. The call generates the “Register.txt” file needed by customer service to generate a user license. It allows the user to view the file contents, and the display of the file can be reset when license keys are exchanged.

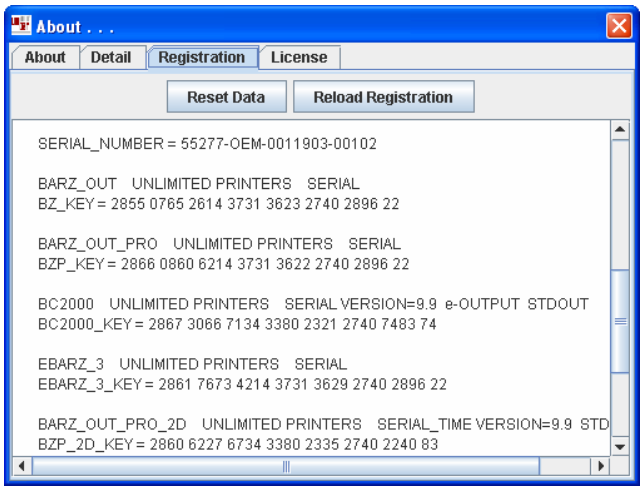

About . . .
About
Detail
Registration
License
Reset Data
Reload Registration
SERIAL_NUMBER = 55277-OEM-0011903-00102
BARZ_OUT UNLIMITED PRINTERS SERIAL
BZ_KEY = 2855 0765 2614 3731 3623 2740 2896 22
BARZ_OUT_PRO UNLIMITED PRINTERS SERIAL
BZP_KEY = 2866 0860 6214 3731 3622 2740 2896 22
BC2000 UNLIMITED PRINTERS SERIAL VERSION=9.9 e-OUTPUT STDOUT
BC2000_KEY = 2867 3066 7134 3380 2321 2740 7483 74
EBARZ_3 UNLIMITED PRINTERS SERIAL
EBARZ_3_KEY = 2861 7673 4214 3731 3629 2740 2896 22
BARZ_OUT_PRO_2D UNLIMITED PRINTERS SERIAL_TIME VERSION=9.9 STD
BZP_2D_KEY = 2860 6227 6734 3380 2335 2740 2240 83

Figure A-3: Provides access to the “Register.txt” file.

License Tab

The forth tab, “License,” allows the user to make the command line call to the formatter using the “-i” switch directly from the User interface, thus eliminating the need to open a separate console. The call generates the “license.txt” file allowing the user to check the contents of their license specifically relating to its expiration date. It allows the user to view the file contents, and the display of the file can be reset when license keys are exchanged.

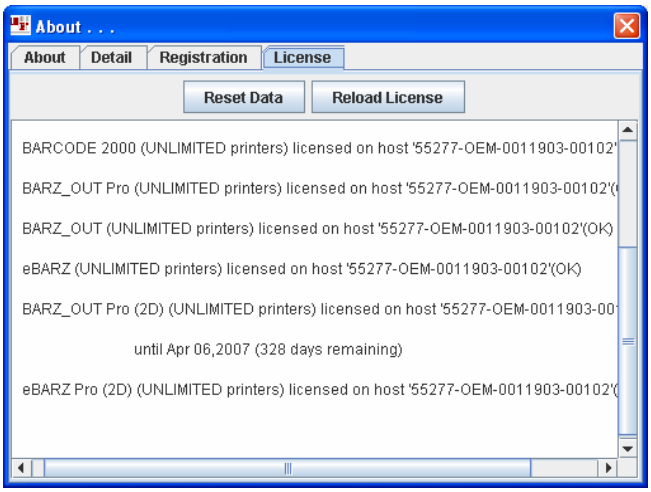
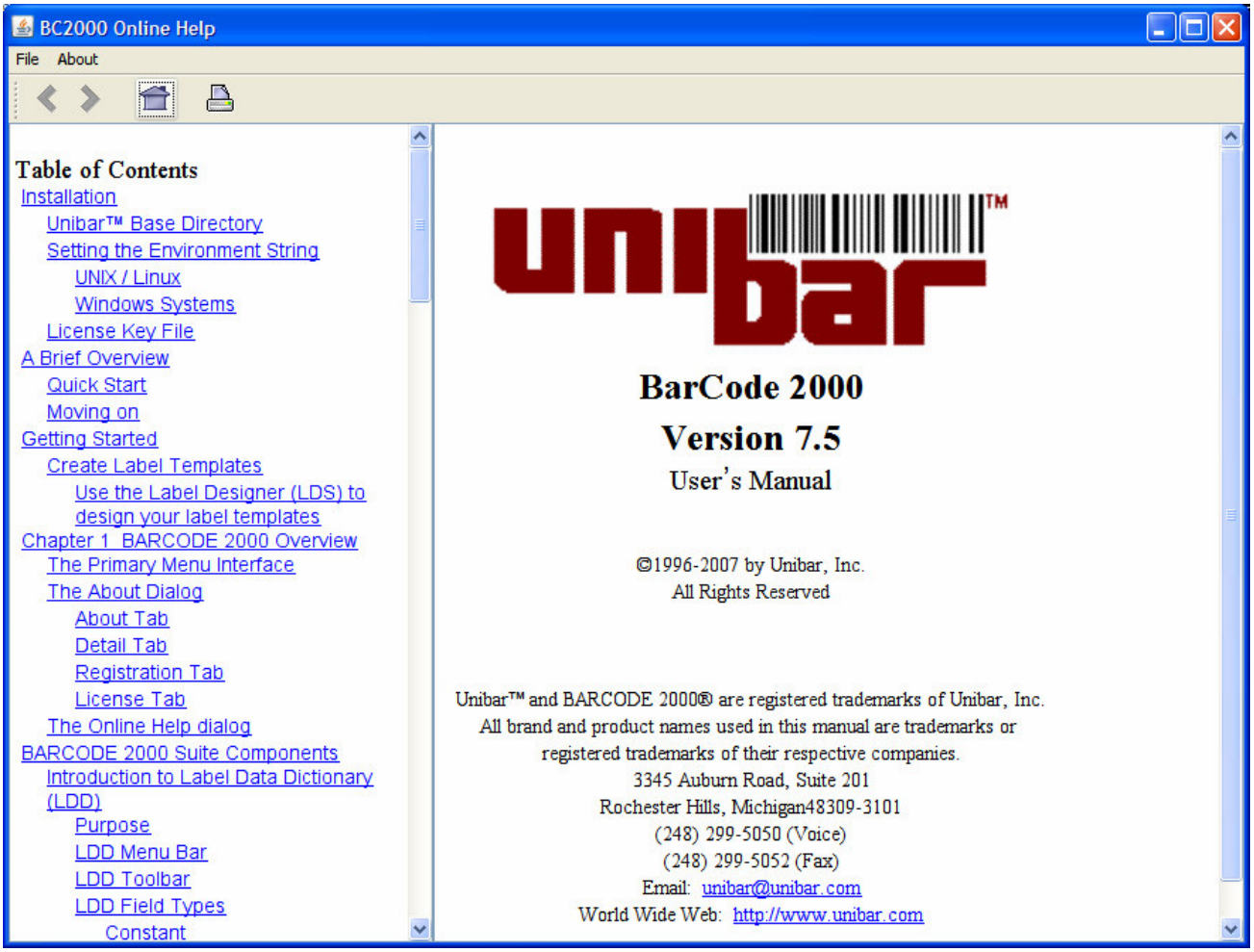

About . . .
About
Detail
Registration
License
Reset Data
Reload License
BARCODE 2000 (UNLIMITED printers) licensed on host '55277-OEM-0011903-00102'
BARZ_OUT Pro (UNLIMITED printers) licensed on host '55277-OEM-0011903-00102'(
BARZ_OUT (UNLIMITED printers) licensed on host '55277-OEM-0011903-00102'(OK)
eBARZ (UNLIMITED printers) licensed on host '55277-OEM-0011903-00102'(OK)
BARZ_OUT Pro (2D) (UNLIMITED printers) licensed on host '55277-OEM-0011903-00
until Apr 06,2007 (328 days remaining)
eBARZ Pro (2D) (UNLIMITED printers) licensed on host '55277-OEM-0011903-00102'(

Figure A-4: Provides access to view license expiration info.

The Online Help dialog

The user may also access the User Manual as an Online-Help module from the Help → Help menu [or the F1 key]. The online help is accessible from any of the four applications and opens to the appropriate page when opened from them.



The online help manual contains a menu driven table of contents [left window] and allows the user the opportunity to print the pages they are viewing [right window]. This help browser keeps track of which pages the user views during any single session. The left and right arrows navigation from where you are to where you have been. The Home button takes the user back to the home page [see above]. The Printer icon prints the right pane contents to the viewing computer's default printer.

BARCODE 2000 Suite Components:

- The Label Data Dictionary (**LDD**): The BARCODE 2000 data source.

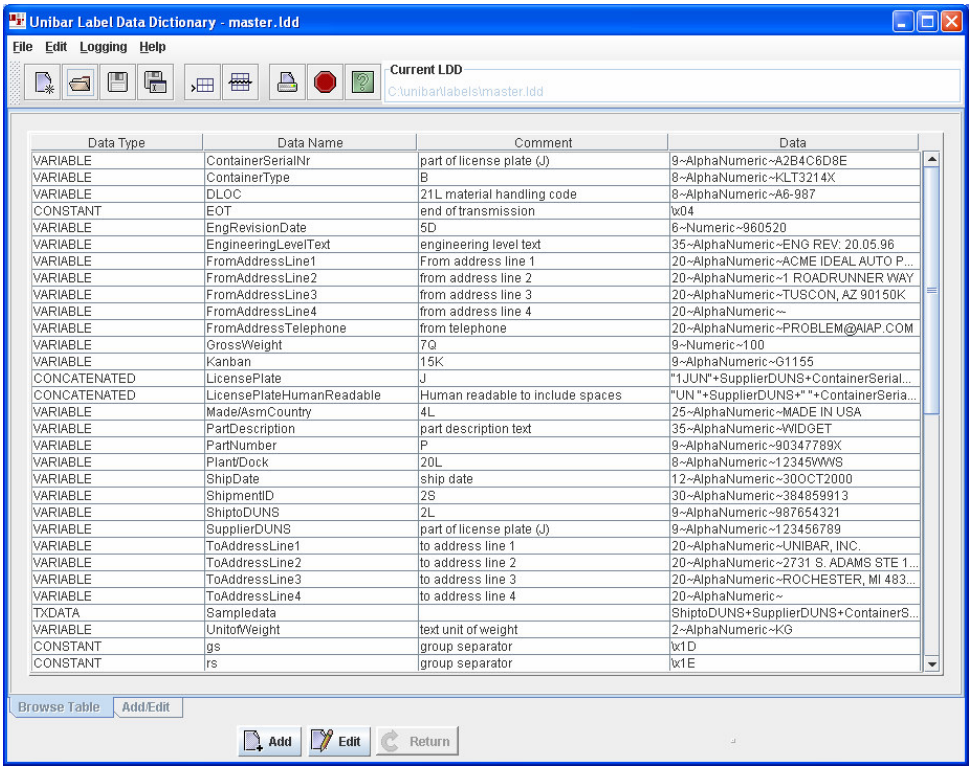

Unibar Label Data Dictionary - master.ldd
File Edit Logging Help
Current LDD
C:\unibar\labels\master.ldd
Data Type | Data Name | Comment | Data
VARIABLE | ContainerSerialNr | part of license plate (J) | 9~AlphaNumeric~A2B4C6D8E
VARIABLE | ContainerType | B | 8~AlphaNumeric~KLT3214X
VARIABLE | DLOC | 21L material handling code | 8~AlphaNumeric~A6-987
CONSTANT | EOT | end of transmission | \x04
VARIABLE | EngRevisionDate | 5D | 6~Numeric~960520
VARIABLE | EngineeringLevelText | engineering level text | 35~AlphaNumeric~ENG REV: 20.05.96
VARIABLE | FromAddressLine1 | From address line 1 | 20~AlphaNumeric~ACME IDEAL AUTO P...
VARIABLE | FromAddressLine2 | from address line 2 | 20~AlphaNumeric~1 ROADRUNNER WAY
VARIABLE | FromAddressLine3 | from address line 3 | 20~AlphaNumeric~TUSCON, AZ 90150K
VARIABLE | FromAddressLine4 | from address line 4 | 20~AlphaNumeric~
VARIABLE | FromAddressTelephone | from telephone | 20~AlphaNumeric~PROBLEM@AIAP.COM
VARIABLE | GrossWeight | 7Q | 9~Numeric~100
VARIABLE | Kanban | 15K | 9~AlphaNumeric~G1155
CONCATENATED | LicensePlate | J | "1JUN"+SupplierDUNS+ContainerSerial...
CONCATENATED | LicensePlateHumanReadable | Human readable to include spaces | "UN "+SupplierDUNS+" "+ContainerSeria...
VARIABLE | MadeAsmCountry | 4L | 25~AlphaNumeric~MADE IN USA
VARIABLE | PartDescription | part description text | 35~AlphaNumeric~WIDGET
VARIABLE | PartNumber | P | 9~AlphaNumeric~90347789X
VARIABLE | PlantDock | 20L | 8~AlphaNumeric~12345WWS
VARIABLE | ShipDate | ship date | 12~AlphaNumeric~30OCT2000
VARIABLE | ShipmentID | 2S | 30~AlphaNumeric~384859913
VARIABLE | ShiptoDUNS | 2L | 9~AlphaNumeric~987654321
VARIABLE | SupplierDUNS | part of license plate (J) | 9~AlphaNumeric~123456789
VARIABLE | ToAddressLine1 | to address line 1 | 20~AlphaNumeric~UNIBAR, INC.
VARIABLE | ToAddressLine2 | to address line 2 | 20~AlphaNumeric~2731 S. ADAMS STE 1...
VARIABLE | ToAddressLine3 | to address line 3 | 20~AlphaNumeric~ROCHESTER, MI 483...
VARIABLE | ToAddressLine4 | to address line 4 | 20~AlphaNumeric~
TXDATA | Sampledata | | ShiptoDUNS+SupplierDUNS+ContainerS...
VARIABLE | UnitofWeight | text unit of weight | 2~AlphaNumeric~KG
CONSTANT | gs | group separator | \x1D
CONSTANT | rs | group separator | \x1E
Browse Table
Add/Edit
Add
Edit
Return

Figure 1-A: Easily define your data format using our Label Data Dictionary (LDD) component

- The Label Design System (**LDS**): The label format creation utility.

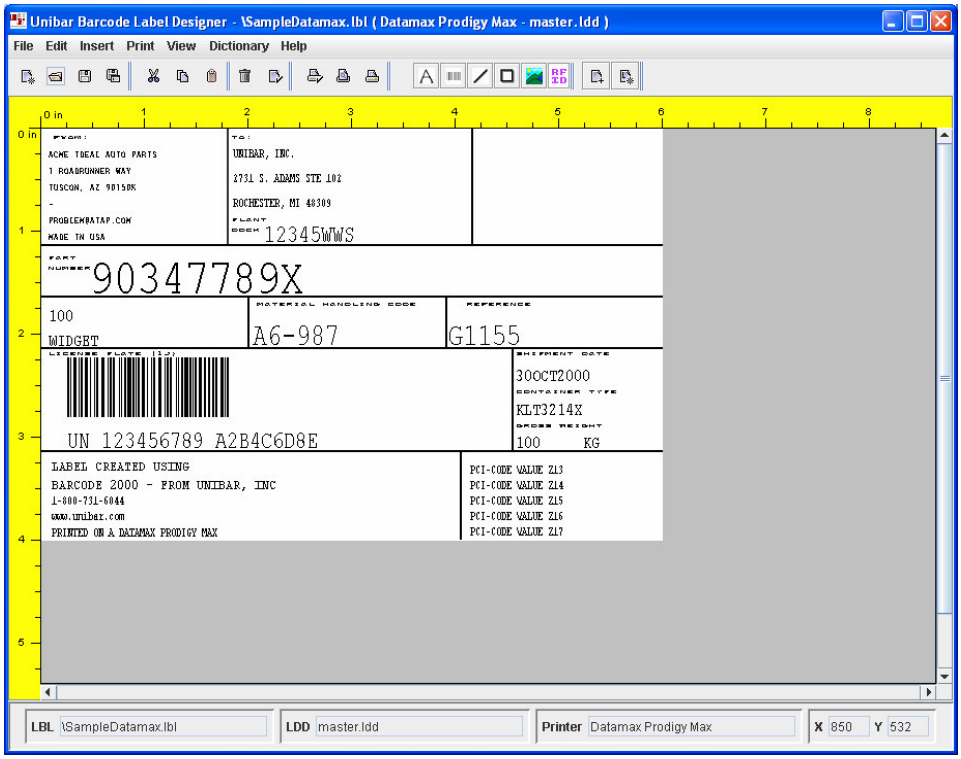


Figure 1-B: Point, Click, Drag and Drop any of the fields on the label

- The Interactive Data Entry Environment (**IDE**): The test bed and sample file printing utility.

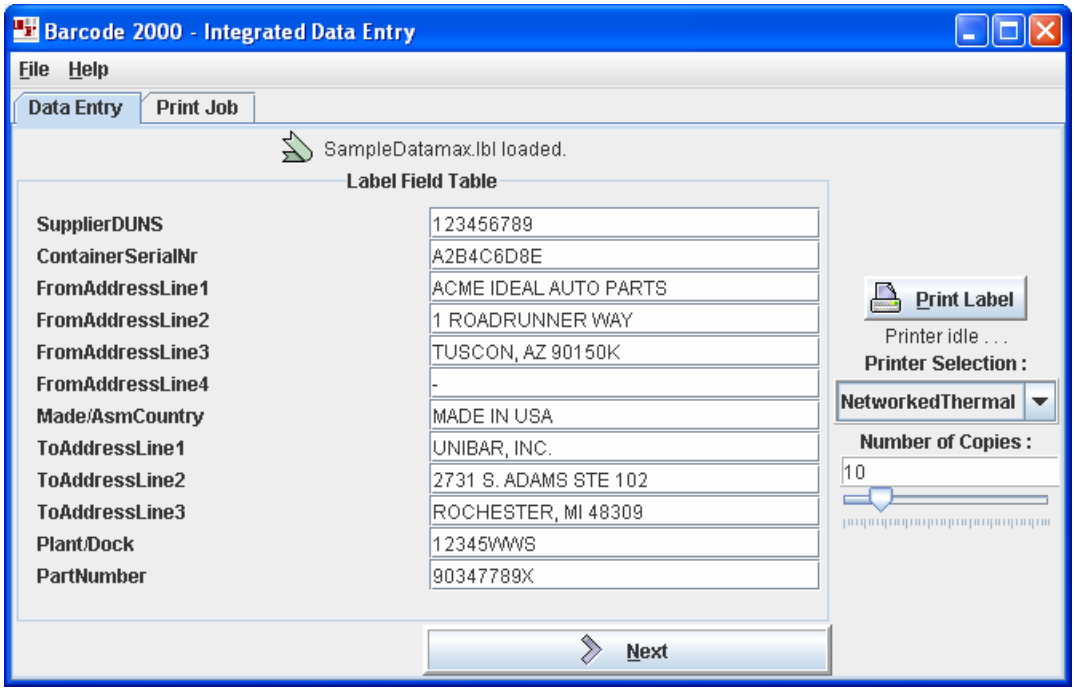


Figure 1-C: Test print or print labels one at a time or in batches with our Integrated Data Entry Environment (IDE)

- The Printer Configuration Tool

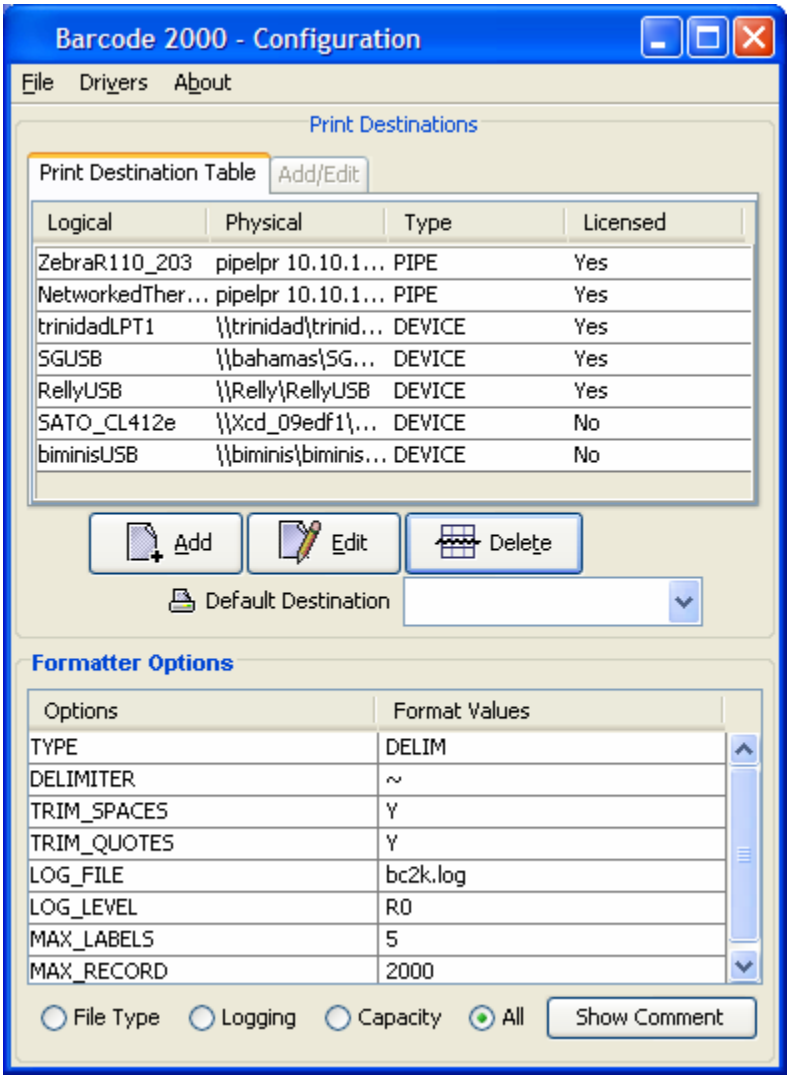


Figure 1-D: Setup printer destinations easily in our Configuration Tool.

Introduction to Label Data Dictionary (LDD)

Purpose

The LDD stores all your data variables, concatenations and other special fields. The field type determines the operations to be preformed by the Formatter when printing. In order for you to use a variable in a label template, it must be defined in the LDD you have created for the label template.

Note: You can create a single LDD for each label template or create a global LDD that can be used with all your label templates.

LDD Menu Bar

The menu bar is located at the top of the LDD main window. This menu bar is similar to other menu bars found in most Windows and Java applications.

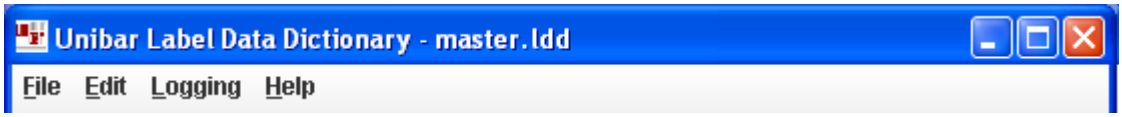
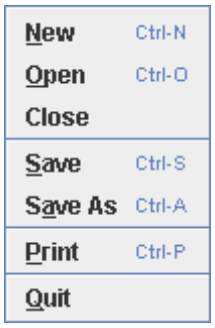


Figure 1-E: LDD Menu Bar

File Menu



- Create New LDD file
- Open existing LDD file
- Close the current LDD file
- Save the current LDD file
- Print contents of the LDD to the default system printer
- Quit the application

Edit

- Copy and append LDD contents to another LDD file
    - Copy to either a new or existing LDD
- Delete a single field
    - Deleted fields may be restored prior to saving the LDD

Logging Display errors that may occur when loading/saving LDD files

Help

- Access Quick-Help to find help regarding LDD operations
- Access the Online Help window
- Access the About/Information Dialog

LDD Toolbar

The toolbar provides quick and easy access to commonly used menu commands. The toolbar is located below the menu bar and is removable is needs be. When the corner "X" is clicked it reverts to is origin on the dialog. Not that the current open LDD is displayed at the right.

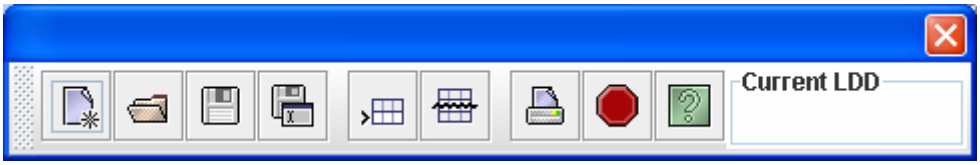


Figure 1-F: LDD Toolbar

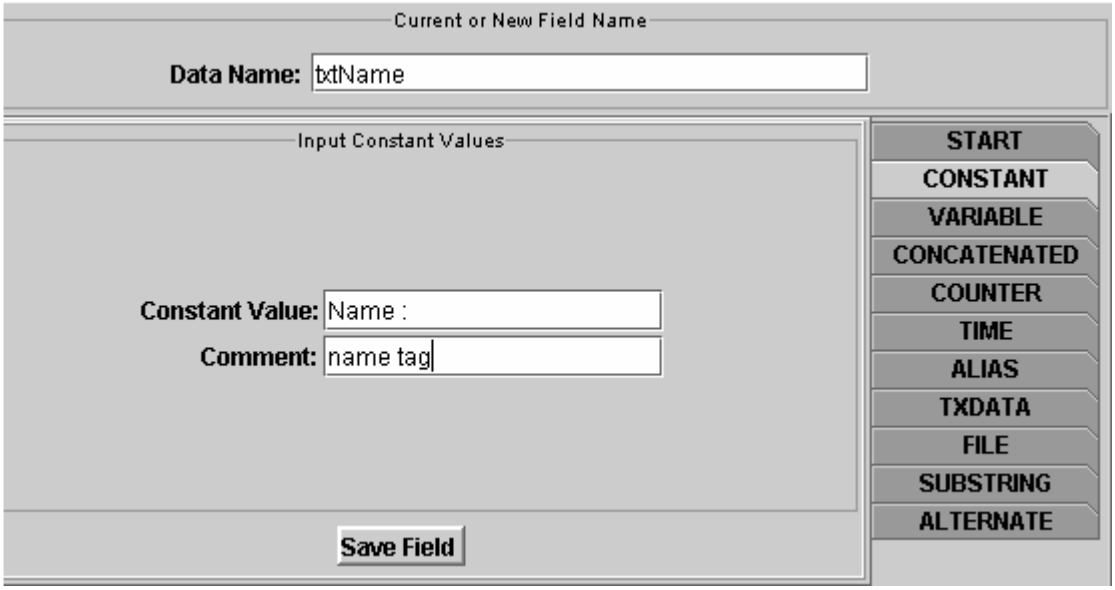
New		Opens a new blank LDD file
Open		Opens an existing LDD file
Save		Saves the current LDD to a file
Save As		Saves the current LDD file to a different file name
Append		Appends selected row(s) into another file or a new file
Delete		Deletes the selected row from the LDD
Print		Prints the current LDD to the default system printer
Quit		Closes the current LDD file, while leaving the interface open
Quick Help		Displays short detailed information about the LDD application
Current LDD		Displays the full path of the current LDD on the system

LDD Field Types

Constant

The constant field is a static field. The data in the 'Constant Value' field will be printed on the label at print time.

Note: Constants can also be defined in the LDD rather than using TEXT fields in the label template. This makes the constant field consistent in all label templates. It can be changed in the LDD and will immediately be reflected in all label templates using the field.



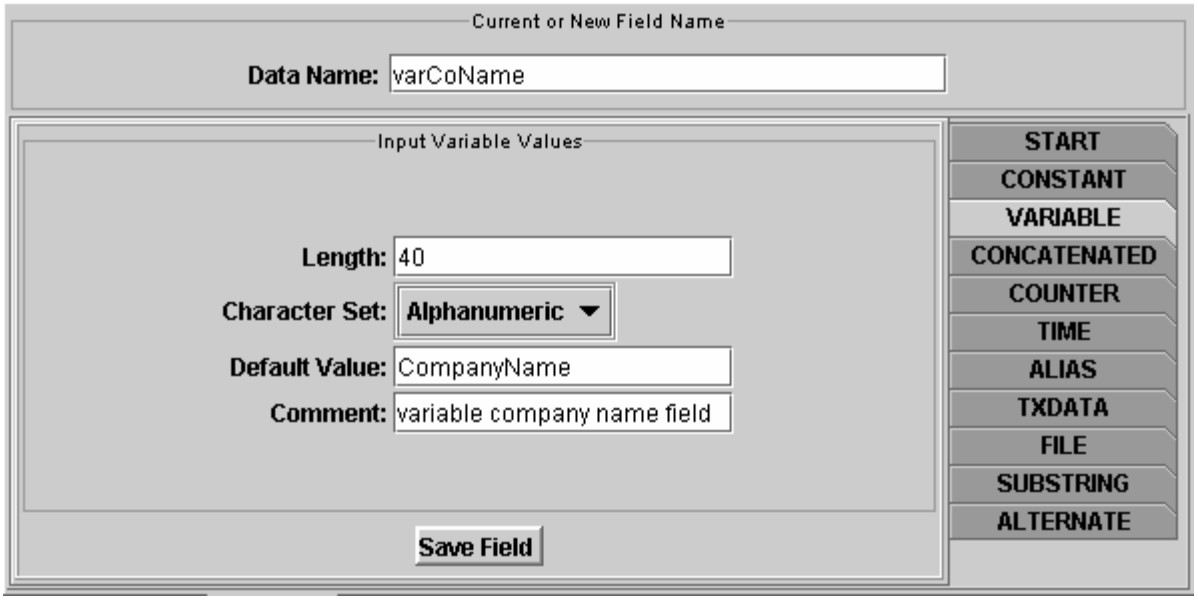
Data Name	A unique name for the field
Constant Value	An alphanumeric value or binary
Comment	A comment about the data in the field

Variable

This field defines data that changes at print time. An example of a variable field would be the weight on a shipping label that changes for each box. LDD variables can also be added to the current dictionary while using the Label Design System (LDS). The user interface is somewhat different but the data input by the user is the same.

Although new fields can be created elsewhere, variables and other dictionary items may only be edited (that is changed) from within the LDD editor itself. Field names are not editable.

Variable-values may be generated from other sources – such as spreadsheets and databases. Instructions for utilizing data from Excel® and Access® are explained in Appendix E.



Data Name A unique name for the field

Length Maximum length of the field. *If the data exceeds the maximum length set, the remainder of the field will be truncated.*

Character Set Sets the character set of the variable.
Alphanumeric – Characters and digits – i.e. the SQL VARCHAR
Binary – Binary digits designated by a "0b" prefix
Hexadecimal – Hexadecimal digits designated by a "0h" prefix
Numeric – Digits only

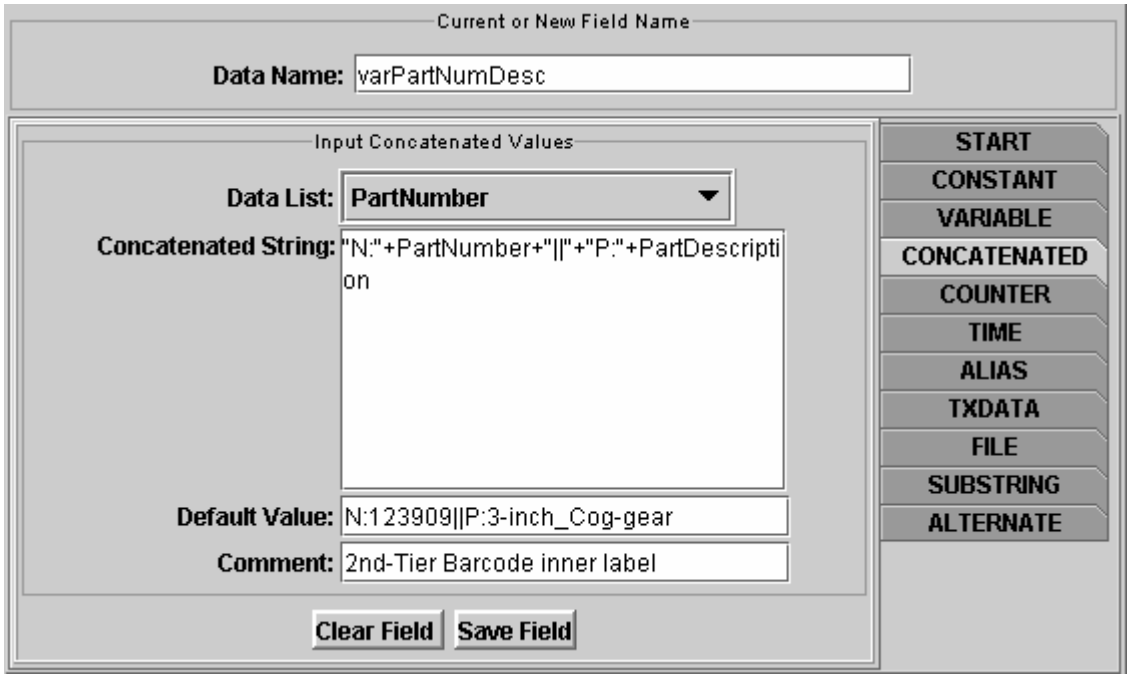
Default Value Sets a default value that is used only when test printing a label template in the LDS and when printing from the IDE.

Comment A comment about the data in the field

Concatenated

This field is used to combine one or more fields into one single field. An example of using the concatenated fields would be adding 420 to a zip code on a label.

Concatenated strings are usually composed of existing LDD constants and/or variables. However, extra text may be added directly into the string editor pane. Using the drop-down data list pre-appends a "+" to the C-String. If you wish to add your own text, ONLY add the prefix "+" not as suffix. This may cause errors reading the file by the Label Designer (LDS). The LDD editor will simply ignore the extra "++", but the label-format using the C-String will fail to open.



Data Name A unique name for the field

Data List A list of fields you can use to create the concatenated string

Concatenated String Displays the concatenated string.
Note: *You can directly edit the fields and constant data in this area. Using quotes around the data is assumed to be constant.*
Example: "12345"+MyField1+"ABCDE"+MyField2
MyField1 and MyField2 are variable fields setup in the LDD.
Example:
A part number with an identifier of 'P' added to the 'PartNumber' variable field would look like "P"+PartNumber.

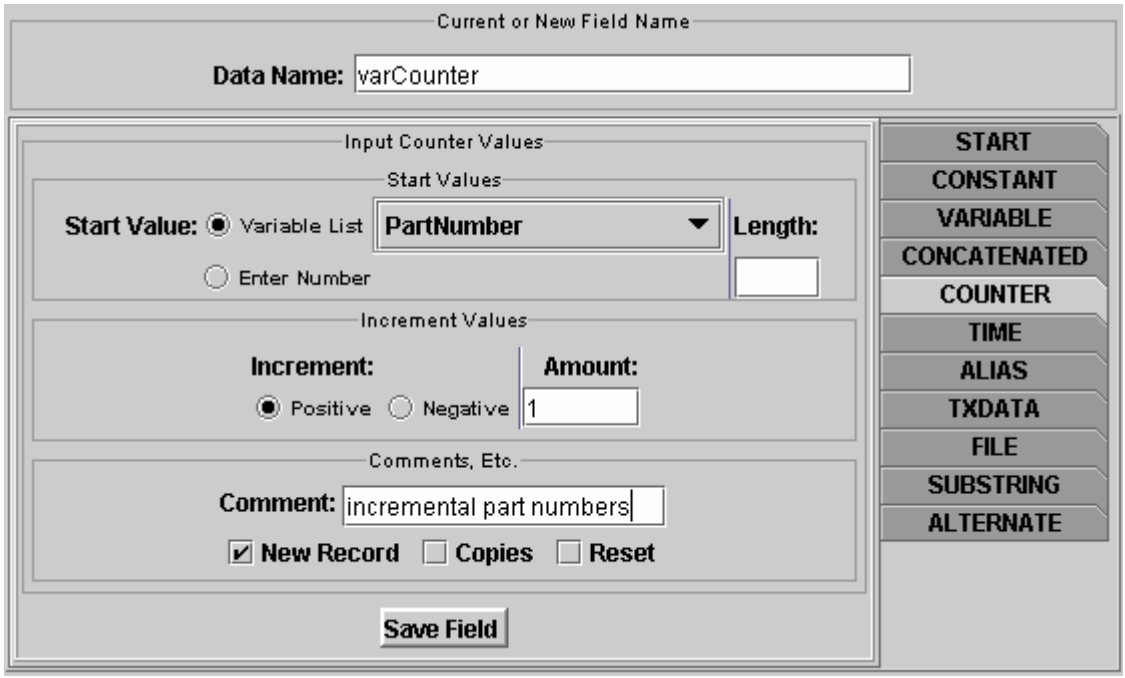
Default Value Sets a default value that is used only when test printing a label template in the LDS and when printing from the IDE.

Comment A comment about the data in the field

Counter

This field is used to automatically increment/decrement a value of a barcode or text field on the label template.

For example, suppose an order consists of a set of three cartons, which are numbered 200, 201, and 202. To generate a set of labels, you set the starting variable to 200 and increment the value by one and print three copies of the same label. Everything on the label is the same for each label except the counter field.



Data Name A unique name for the field

Start Values

Variable List Choose the field that will have the starting value for the counter.

Enter Number Enter the starting number of the counter. The counter will start at this number each time you print the label(s).

Length Specifies the maximum length for the counter. The full length is used; counters that do not fill the entire length are left padded with 0's.

Increment Values

Positive/Negative Increment or decrement the counter field

Amount Sets the amount to increment/decrement the value specified in the Start Values section.

Comments, Etc.

Comment A comment about the data in the field

New Record Increments/decrements on a new record

Copies Increments/decrements on a copy of a record

Reset Resets the counter field

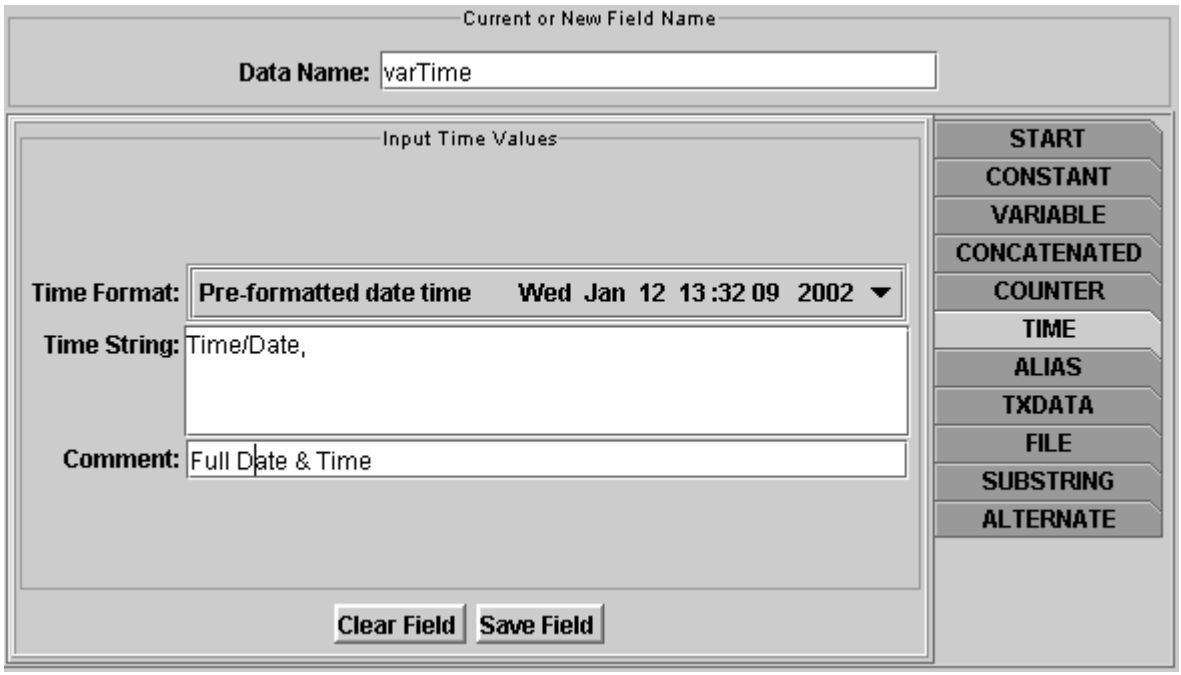
Examples: *(each line is a record)*
Checking Copies and Reset and printing 3 copies of each record would produce the following results
”Box 1 of 3”, “Box 2 of 3”, “Box 3 of 3”
”Box 1 of 3”, “Box 2 of 3”, “Box 3 of 3”
”Box 1 of 3”, “Box 2 of 3”, “Box 3 of 3”

Checking only Copies would increment/decrement on each label printed.

Checking only New Record would increment/decrement on each record
SN00123, SN00123, SN00123
SN00124, SN00124, SN00124
SN00125, SN00125, SN00125

Time

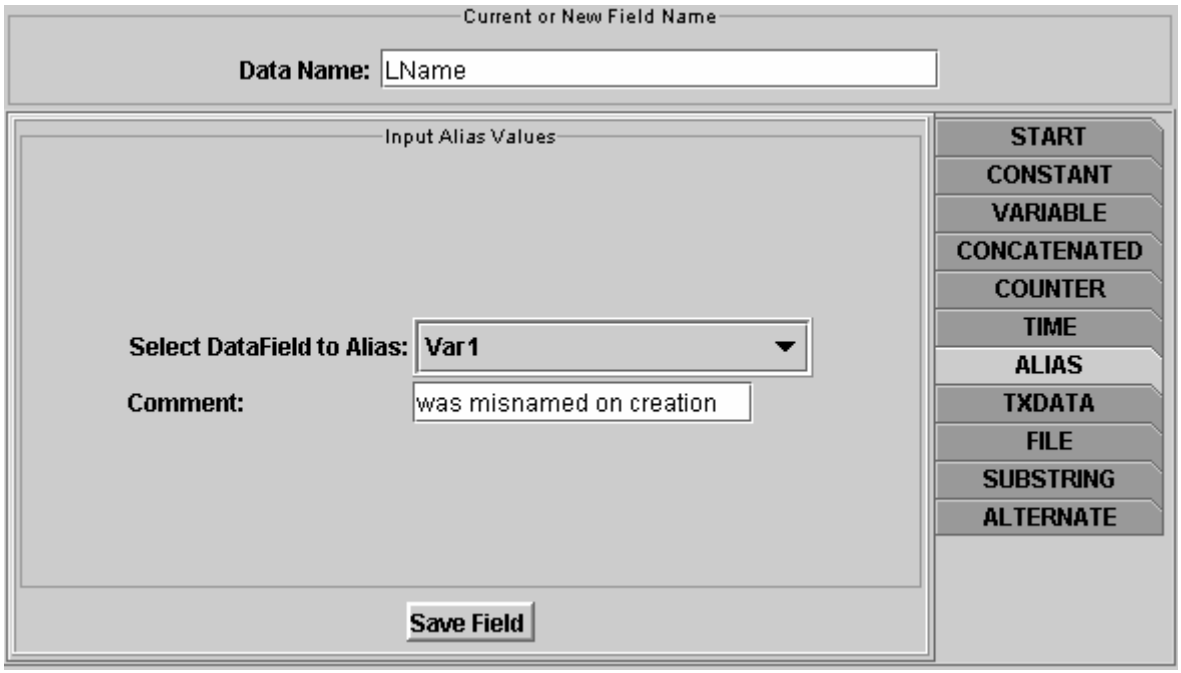
The time field defines a string that you can use in the label template. The string can be a date, time or both depending on how you format it. The Formatter, at print time, will retrieve the current system time and date and format that information into the time string you defined.



Data Name	A unique name for the field
Time Format	A list of time and date formats that can be used to build the Time String
Time String	Displays the current format of the date and time for the field
Comment	A comment about the data in the field

Alias

An alias is just another way to represent a field already defined in the LDD. This can be useful for mapping a field in a database to a field in the LDD without changing the original LDD field name.



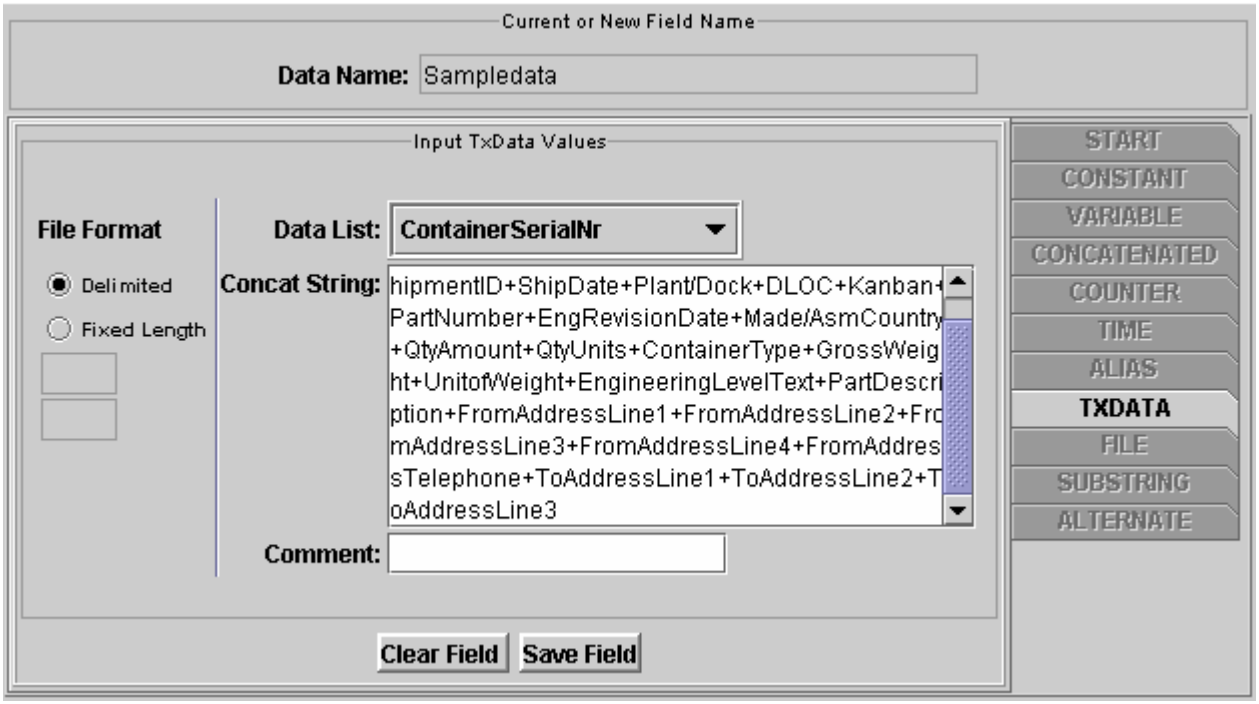
Data Name	A unique name for the field
Select Data Field to Alias	A list of fields you can use to create an alias to map to.
Comment	A comment about the data in the field

TXDATA

This field defines a map of your data that will be sent to the Label-Formatter. Since, there are many ways to format your data; Unibar™ has created this TXDATA field to meet most of our customers' file formats.

A Data Record Format (DRF) can be defined in the Label Data Dictionary. The field type is "TxData" (for Transaction Data Record Format). In the LDD program, you specify a TxData name (DR01, if you're not too creative). This name is used in the data stream to refer to the DRF. You can add existing variable names to the TxData and specify their offset and length if you choose a fixed-length field type of input.

The IDE uses a specific naming convention: “TxData.” + label name + “.ide” to read a DRF.



Data Name — A unique name for the field

File Format

Delimited/Fixed — Choose either use a delimited or fixed data. Delimited files have some sort of field separator (field1~field2~field3). In a fixed data file, the data is in a fixed position and never changes from record to record (field1field2field3).

Offset – (Fixed) — Sets the position of the start of the field in a fixed data file.
Example: data=100120023003
An offset of 5 and a length of 4 would equal ‘2002’

Length – (Fixed) — Sets the length of the field starting from the offset.
(See example in ‘Offset’)

Other

Data List — A list of fields you can use to create the concatenated string of the TXDATA field

Concat String — Displays the format of the fields used in the TXDATA
Example:
Delimited Format – Field1+Field2+Field3
Fixed Format – Field1:0:5+Field2:6:7+Field3:14:5

Comment — A comment about the data in the field

Following is a method that provides the Label-Formatter the capability of finding your TxData without specifying it in your data file. This was originally provided for Version 3 users who do not make application changes to upgrade to Version 6.

1) LDD must be named "<*label_name*>.ldd"; TxData is named TxData.<*label_name*>"

2) Use one TxData per label or use a master TxData, named DEFAULT_TXDATA.

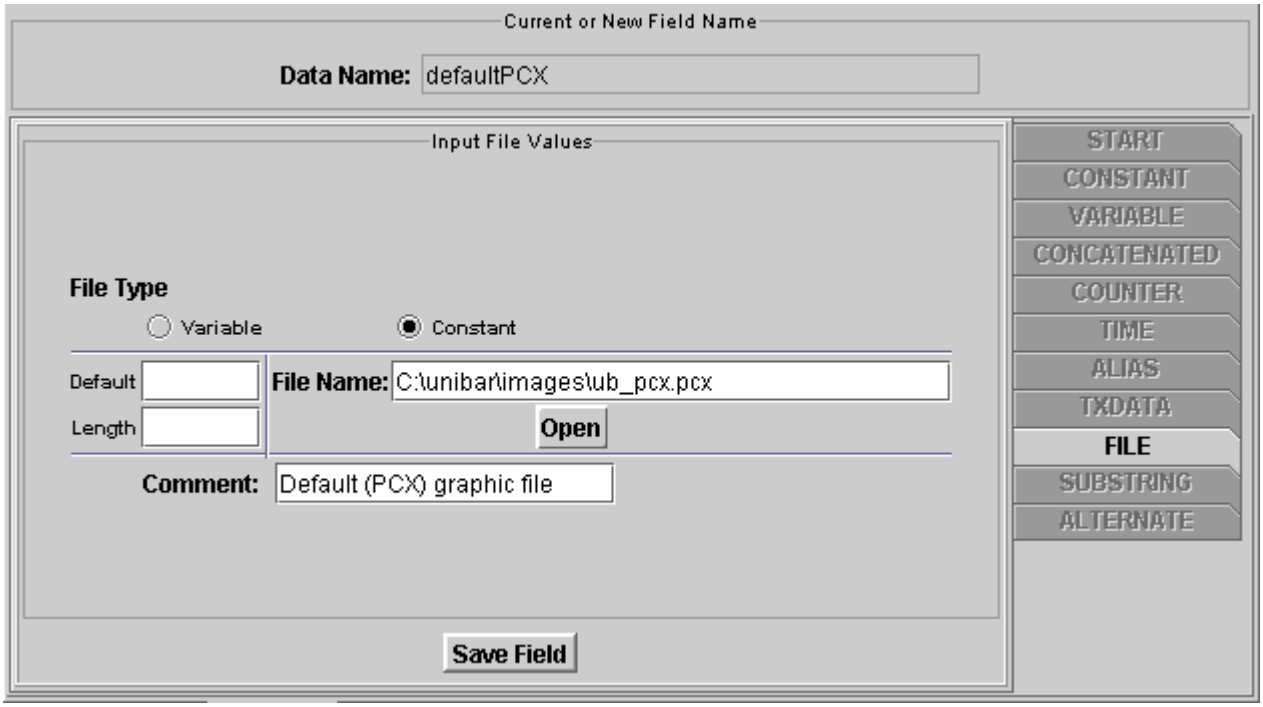
3) UBFMT: if no '/DR' is provided (i.e. No TxData), use DEFAULT_TXDATA if present, else look for a TXDATA named "<*label_name*>"

File – Graphic Image storage

This field defines a graphic file that can be used on a label template. Barcode 2000 allows for variable graphics or constant graphics to be used on the label templates. Please note that if you design on a Windows machine, then move to a non-Windows production environment, the file location will need to be updated. For example, C:\unibar\images\ub_pcx.pcx may need to be updated to /usr/unibar/images/ub_pcx.pcx.

A default graphic image of type PCX and JPG come delivered with Barcode 2000 in an image sub-directory. They should be considered FILE-Type constant. They exist for the "Quick Label" demonstration.

The FILE-Type "variable" would consist of a group of graphic files that varies through a run of labels. See below for an explanation of the settings.

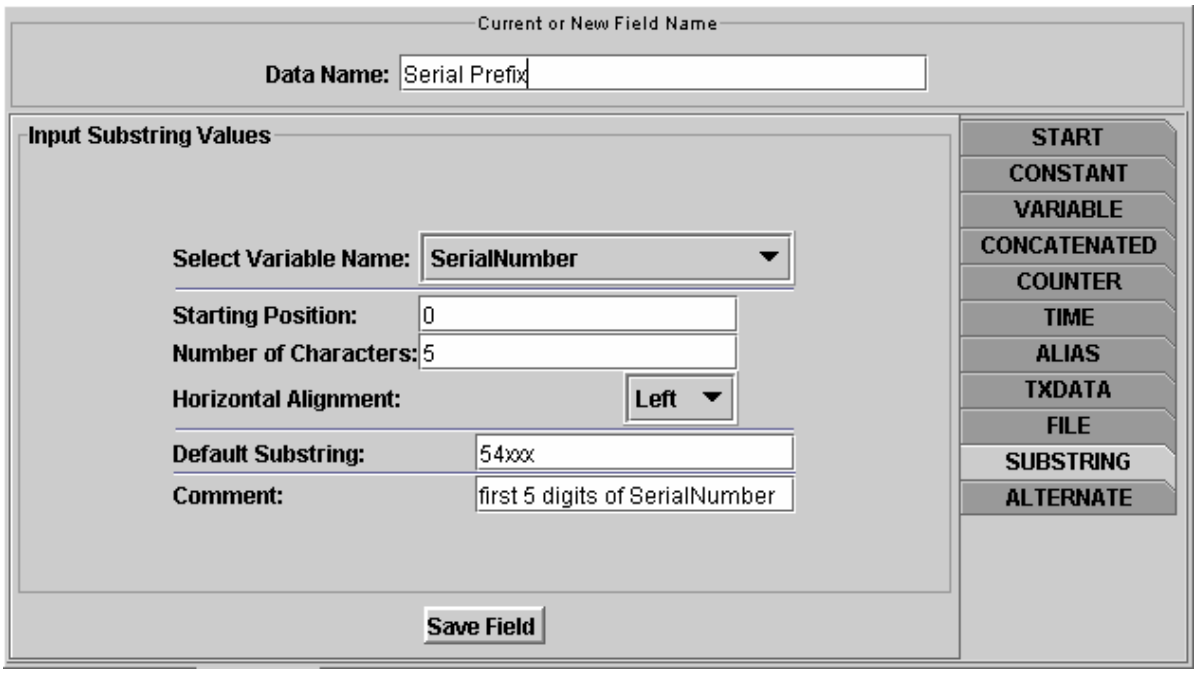


Data Name A unique name for the field

Variable/Constant	Sets the type of file you are using for this field.
File Name – (Constant)	Sets the path of the file. *Use the absolute path.* **Example:** C:\mygraphics\pic.pcx or /usr/mygraphics/pic.pcx
Default – (Variable)	Sets a default value that is used only when test printing a label template in the LDS and when printing from the IDE.
Length – (Variable)	Maximum length of the field. *If the data exceeds the maximum length set, the remainder of the field will be truncated. For this field type we recommend making the length 50 or greater, because of the variances in file names that might be used.*
Comment	A comment about the data in the field

Substring

This field is used to extract a portion of data from an existing LDD field. An example of a use of the substring would be to extract the day of the month from a know format of 'mm/dd/yyyy'. In which case, you would have a LDD field defined and the field would contain 'mm/dd/yyyy'. To extract the day of the month you would set the Starting Position to 4, Number of Characters to 2 and Horizontal Alignment to Left.

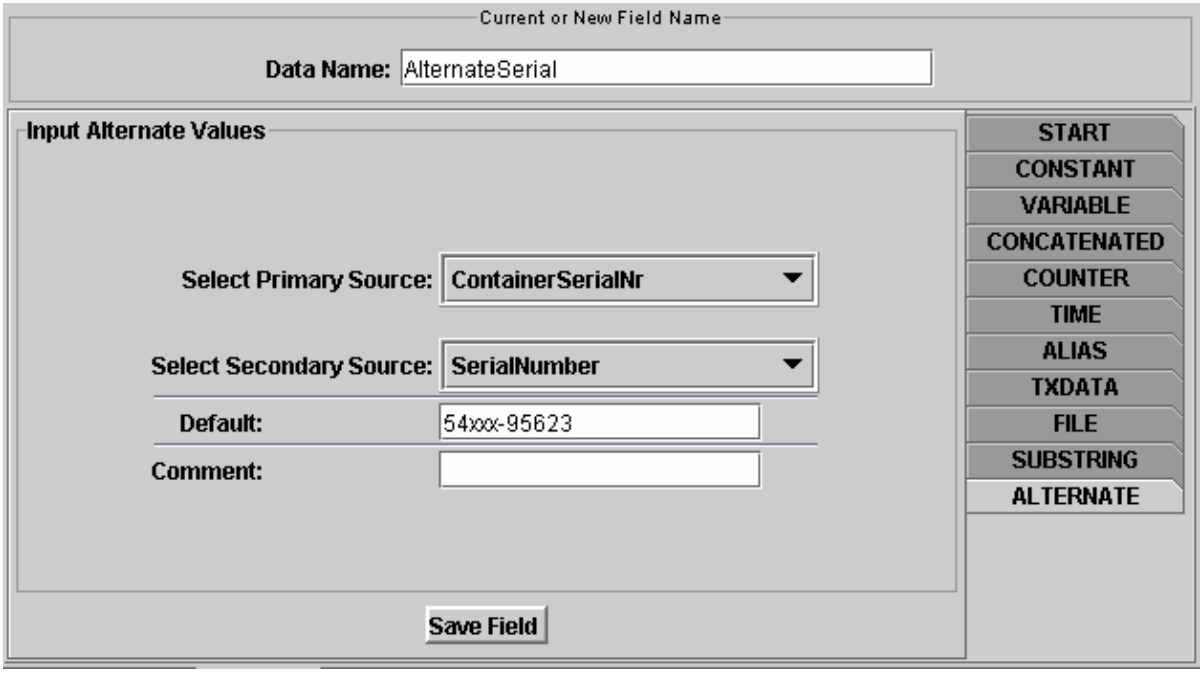


Data Name	A unique name for the field
Select Variable Name	A list of fields you can use to create the substring field
Starting Position:	The starting point of where the substring is to start

Number of Characters	The number of characters to use in the substring field from the Starting Position
Horizontal Alignment	Determines where the Starting Position is to start from. **Example:** If Horizontal Alignment were set to Right then the starting point of the substring would be the value in the Starting Position counting from the right.
Default Substring	Sets a default value that is used only when test printing a label template in the LDS and when printing from the IDE
Comment	A comment about the data in the field

Alternate

This field provides a function of replacing the primary source with the secondary source only if the primary source is empty



Data Name	A unique name for the field
Select Primary Source	A list of fields you can use for the Primary Source
Select Secondary Source	A list of fields you can use for the Secondary Source when the Primary Source is empty
Default	Sets a default value that is used only when test printing a label template in the LDS and when printing from the IDE
Comment	A comment about the data in the field

For example, suppose you have fields defined as RetailPrice (primary) and SalePrice (secondary). If the RetailPrice were empty (blank), then the SalePrice would be used.

Introduction to Label Design System (LDS)

Purpose

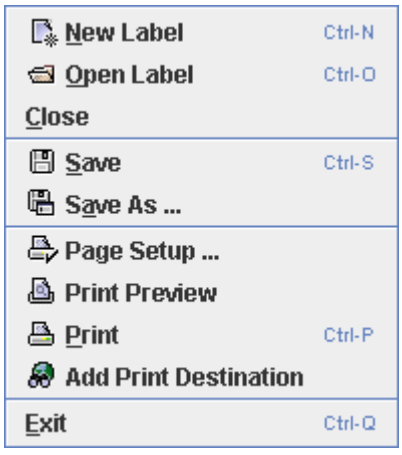
This is where you would define the specifics of the label, such as printer model, speed, print orientation, etc. This is also the component that you define your label template by placing fields (barcodes, text, lines, variables and graphics) on the screen.

Menu Bar

The menu bar is located at the top of the LDS main window. This menu bar is similar to other menu bars found in most Windows and Java applications. All LDS functionality is found in the menus of the menu bar. Most of this behavior is mimicked in the toolbars which are removable (customizable) during a single session. Many users choose to remove all the toolbars and the status bar to attain the largest possible viewing area of the design canvas.

Figure 1-G: LDS Menu Bar

File



- To create new label formats
- To open and close label formats
- To save or rename (save as) label formats
- To print a sample label;
    - set printer destination
    - set printer parameters
    - preview current label template
- To quit the application

Edit

Note: This menu is available via right-click (opposite-click) on the canvas

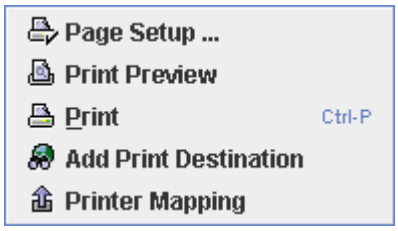
- Edit fields
    - To cut label objects
    - To copy label objects
    - To paste label objects
- Select and Deselect Label objects
- Delete Label objects

Insert

Note: This menu is available via right-click (opposite-click) on the canvas

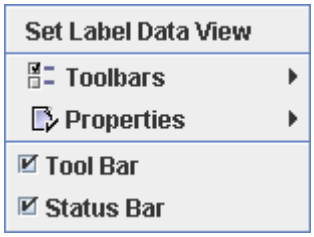
- To insert the basic Label objects
    - Text
    - Barcode
    - Line, Rectangle
    - Graphic
    - Rfid Tags

Print

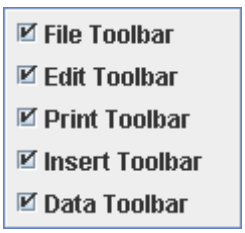


- Set page parameters
- Preview current label template prior to printing
- To print a sample label
- Add a printer destination
- Map the current label format to a different printer

View



- Set the label format data view
- To customize the toolbar
- See label properties
- Toggle the header panel (toolbar) and the footer panel (status bar)



- File: Contains ***New, Open, Save, Save As***
- Edit: Contains ***Cut, Copy, Paste, Delete, Label/Object Properties***
- Print: Contains ***Page Setup, Print Preview, Print***
- Insert: Contains ***an icon to insert each of the six label objects***
- Data: Contains ***Add LDD Entry, Create LDD***

Dictionary

- To add variables to the Label Data Dictionary (LDD) attributed to the current template (Label Design Document).
- Create a new (empty) Label Data Dictionary (LDD)

Help

- To access this online help manual.
- To access the System's information

LDS Toolbar

Figure 1-H: LDS Toolbar

The toolbar provides quick and easy access to commonly used menu commands. There are currently five (5) toggling toolbars – File, Edit, Print, Insert, and Data – they can be turned 'on' & 'off'; The *Insert* toolbar is detachable from the main bar for easy label making and designing.

FILE Section — The File menu and toolbar share four basic functions*: New, Open, Save*, and *Save As*.

New Label Creates a new blank label template

Open Label Opens an existing label template

Save Saves the current label template to a file

Save As Saves the current label template with a different file name

EDIT Section — The Edit Menu and toolbar share three basic functions: *Cut, Copy, and Paste;* additional functions include Delete, and Deselect:

Note both the EDIT Menu and the INSERT menus are available as a right-click Menu-Items in the Designer.

Cut Field Acts like every other kind of Cut operation – it is the sublime combination of copy and delete – if a Paste operation does not follow a Cut then the Item is lost.

Copy Field Acts like any other kind of Copy – if a Paste operation does not follow a Copy the item is lost by the next action involving a Cut or Copy.

Paste Field Single action paste; must follow a cut or copy; multiple pasting is not supported. The object pasted will be set to the last visual coordinates of the cursor, immediately prior to the call to the Paste function.

MISC Section

Delete Field Deletes any label object from the template – PERMANENTLY, after confirmation.

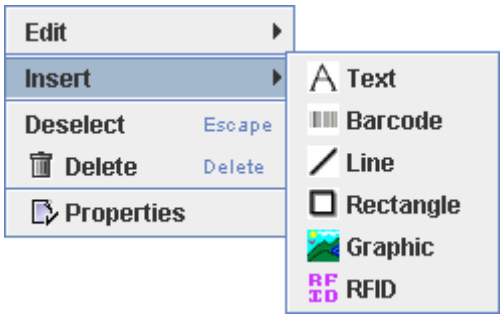
Properties		Gives the label object properties if a label object is selected or the label format properties if no object is selected.
PRINT Section		Based on usage these three print related actions are the most popular
Page Setup		Displays the page setup dialog box
Print Preview		Displays the label template and sets the rotation for the entire label template; the current template can be printed to the current Print Destination from here.
Print		Prints a sample label to the currently selected printer set in the Print Destination Dialog
INSERT Section		Inserts the chosen Label Object at the visual coordinates of the design-space immediately following the button-press of the selected object.
Text		Create a text or data field
Barcode		Create a barcode field
Line		Create a vertical or horizontal line
Rectangle		Create a rectangle
Graphics		Add a graphic
Rfid Tag		Add an Rfid Tag object to the canvas
DATA Section		These two actions allow the user to add to the current Label Dictionary or create a new one.
Add LDD Entry		Add a variable to the current LDD
Add LDD		Create a new “empty” LDD preceding creation of a Label Format

LDS Fields

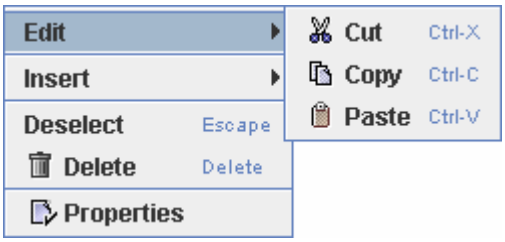
Explanation of Label Object Behavior

Default Label objects consist of Text/Data Strings, Barcodes, Lines, Rectangles, and Graphics. All default label objects can be placed (inserted) on the template via menu bar [**Insert → Object Type**], toolbar, or right-click menu [**Insert → Object Type**] (*see figure 6.1 below*).

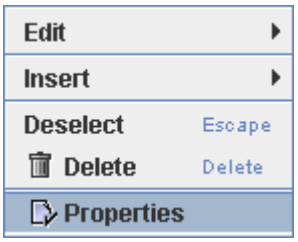
Figures 6.1-6.3 Right-click pop-up menus



Figures 6.1 Insert pop-up menu



Figures 6.2 Editing pop-up menu



Figures 6.3 Properties

Object Selection

To select a label object, move the mouse over the object and click once. A magenta box will form around the object. Once selection is enabled, the selected object can be either moved, edited or deleted.

Object Movement

To move the selected object, press and hold the mouse on the selected object and drag it to the desired position; fine positioning can be attained via a property dialog.

Object Editing

To edit the object, double-click on the object and the object's property dialog will appear. Alternatively, select the object [see above] and either press the properties button on the toolbar , or right-click on the object to raise the pop-up menu and click on the menu's properties selection [*see figure 6.3*].

Object Deletion / De-selection

To delete the selected object, select the item [see above] and either right-click, highlight Delete before releasing the click or press the *Delete-Key* on the keyboard. Additionally, you may also click on the delete button on the toolbar when the object is selected.

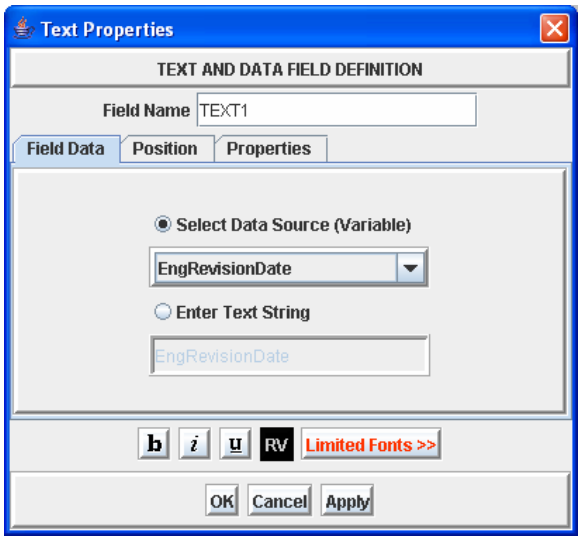
To de-select the item simply press the *Escape-Key* on the keyboard or choose Deselect from the right-click pop-up menu.

Descriptions of Label Objects

Text-Data Field

To add a text field to the label template, Click on the button, and then click on the label template in the vicinity where you want to add the text field. Once the text field is on the canvas, in order to edit that field, click the field to highlight the field, then

Text Dialog – Field Data Tab



Field Name This is a unique identifier for this particular field. Every time the field is added to the label template the number after the field name increases by 1. This is a default behavior.

Example: TEXT1 or TEXT2

New for version 7.5: The field name can be set to a meaningful value by the user.

Text/Data Sets the text field to either static data (text) or dynamic-variable data (data).

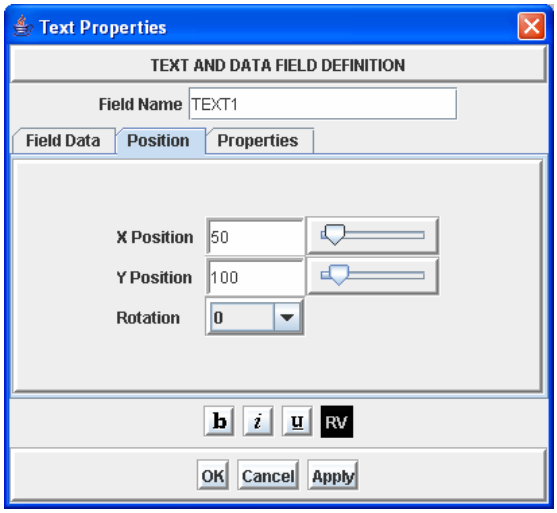
Select Data Source If this radio button is selected, the field becomes a 'Data' field. The user must then use the dropdown box to select a field from the LDD to place on the label template.

Select an LDD variable for your data source

Enter Text String If the field is set to 'Text', then the string value is the static digits/characters you want to print on the label template.

Enter the static text for the field. If the buttons are toggled, the textbox takes the value of the last data-source entry.

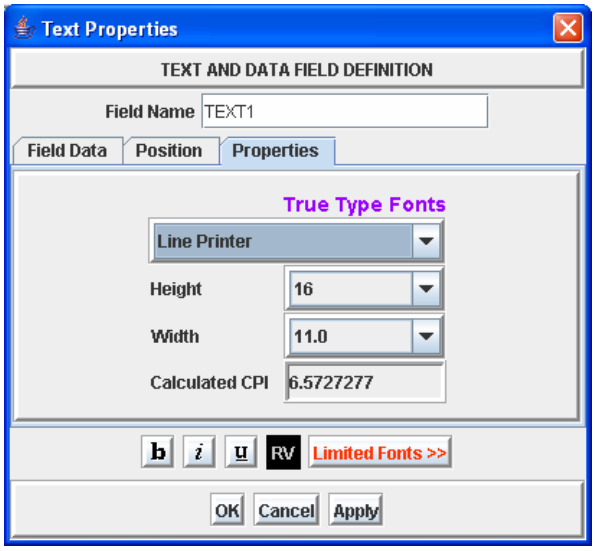
Text Dialog – Position Tab

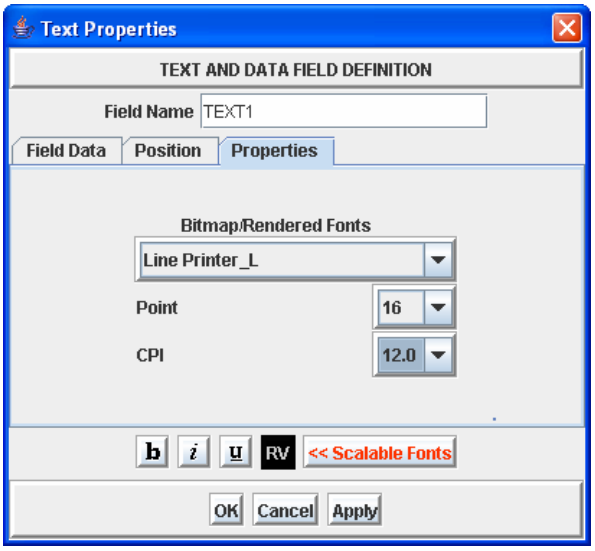


Rotation Sets the degree of rotation for the field.

X Position/Y Position Displays the current location of the Text/Data field using the lower-left corner as the starting point. The units are displayed in 1/100th of an inch (100 equals 1 inch). After the field has been placed on the label template, use this setting to make your fine adjustments.

Text Dialog – Properties Tab





Font Types Sets the font for the particular field. The fonts listed are all the fonts available to the particular printer model for which you are designing.
Note: As of this current release, Barcode 2000 only supports native fonts.

Scalable Fonts Represented by **True Type Fonts** in the designer if available – are pure scalable via Height and Width designations; typically both height and width are represented in points.

Height Sets the Height of each character in points.

Width — Sets the Width of each character in points.

Calculated CPI — Gives the approximate CPI for the given width.

Limited Fonts

Note: LTD fonts are bitmap rendered fonts. They are ugly. They are resident on the printers. They are almost never used but are supported only for backward compatibility. Use them at your own risk!

Represented by a San Serif Family Font in the Designer. The Limited fonts are old and typically inaccurate. The printer manufacturers would prefer you use the newer Scalable Fonts; So do we at Unibar™, Inc.

Note: When selecting the 'Point' size, the CPI that is recommended by Unibar™ is automatically chosen. Fonts that do not support CPI will be set to 'N/A'.

Point — Sets the point size of the font ≈ the height.

CPI (Character Per Inch) — Sets the characters per inch for the font. The larger the number the denser the font will look when printed and vise versa.

Attributes Sets a certain attribute for the field. Any combination can be turned on or off to achieve the desired effect for the text field.

Note: *Grayed out attribute buttons indicate the printer model does not support it.*

Bold — Sets the bold text attribute to either on or off.

Italics — Sets the italics text attribute to either on or off.

Underline — Sets the underline text attribute to either on or off.

Reverse Video — Sets the reverse video text attribute to either on or off.

Barcode Field

To add a barcode field to the label template: Click on the button, [or use the menu **Insert → "Object-Type",** or the Insert Menu from the right-click pop-up menu], then click on the label template canvas in the vicinity where you want to add the barcode field. The default barcode Symbology is *Code 3 of 9* or *Code39* depending on the printer model. When editing the barcode object the following dialog appears:

Barcode Properties – General

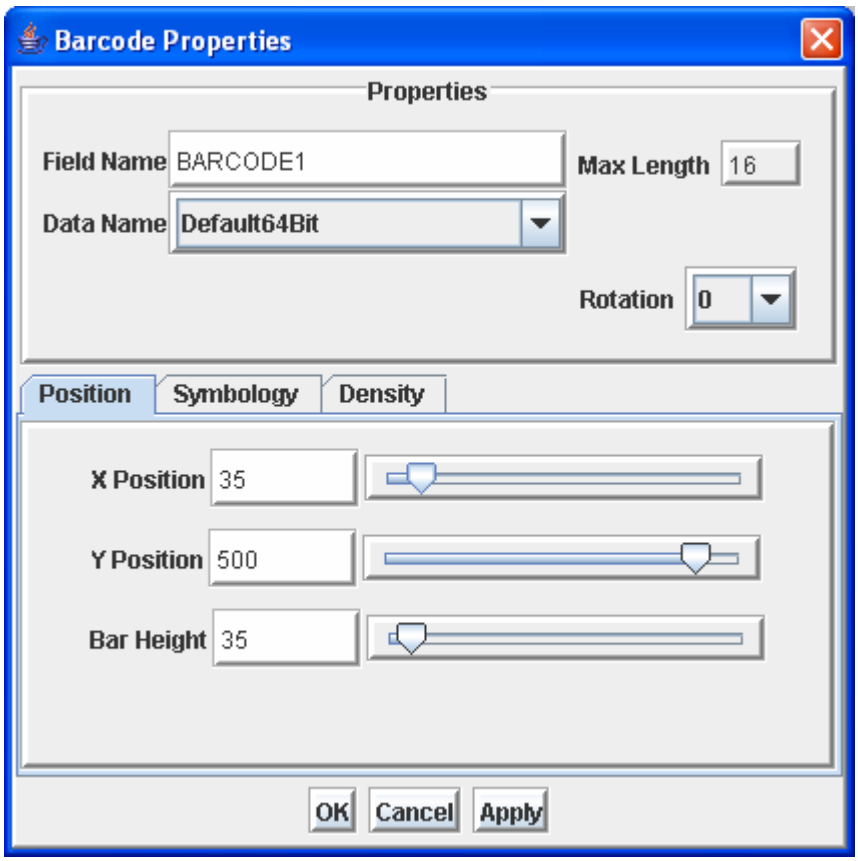
Field Name This is a unique identifier for this particular field.
Example: BARCODE1 or BARCODE2 Every time the field is added to the label template, the number after the field name increases by 1.

Data Name A list of fields you can use for the barcode field from the LDD data source.

Max Length Displays the maximum number of data set for the 'Name' field in the LDD.

Rotation/Orientation Sets the degree of rotation for the field.

Barcode Properties – Position Tab

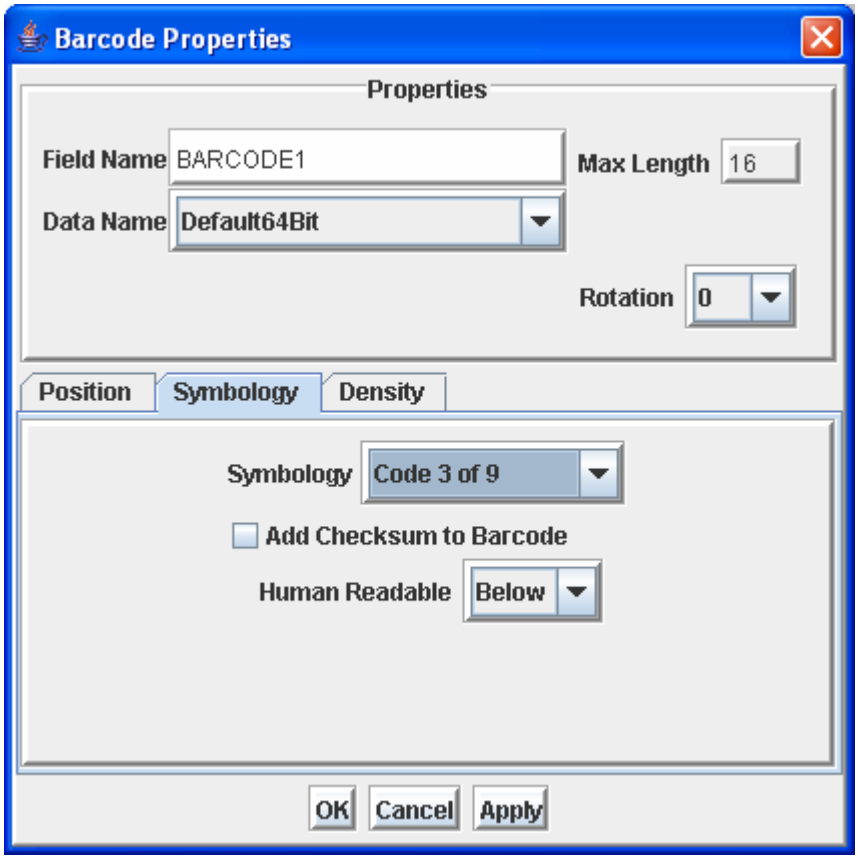


X Position/Y Position Displays the current location of the Barcode field using the lower-left corner as the starting point. The units are displayed in 1/100th of an inch (100 equals 1 inch). After placing the field on the template, use this setting to make your fine adjustments.

Height Sets the height of the barcode.

Note: The units are displayed in 1/100th of an inch (50 equals ½ of an inch).

Barcode Properties – Symbology Tab



Symbology Sets the type of encoding for the field.
Note: The *Symbology List* is printer dependent. A Symbology is only shown in the *Symbology List* if it is supported by the chosen printer model. Hence if the Symbology is NOT in the list, your printer does NOT support that Symbology.

Checksum This sets a property to either include a checksum or not.
The checksum is a character/digit that is included as part of the barcode by a mathematical calculation provided in the specifications for that particular Symbology. It is used to perform a check to ensure the data is read correctly.

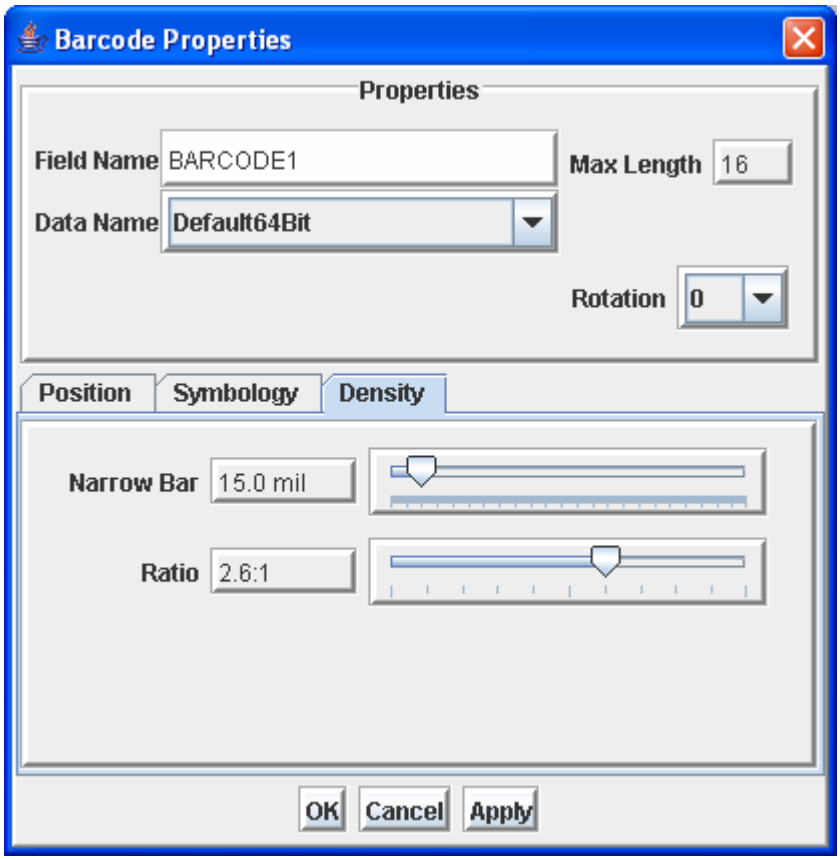
Human Readable This sets how the encoded data is displayed for this field.
Note: This setting is printer dependent. Some printer models do not support human readable as an option for the barcode. In this case we recommend you place a 'Data' field containing the data encoded where you want the human readable to print.

Above An example of a barcode with HR=ABOVE

Below An example of a barcode with HR=BELOW

None An example of a barcode with HR=NONE

Barcode Properties – Density Tab



Narrow Bar, also known as ***Density***

This setting, conjunction 'Ratio', determines the width of the barcode. The setting is displayed in mils. The lower the number (5.0 mils) the smaller the barcode. Range (5.0 to 120.0, .5 increments).
Note: The setting is dependent on the DPI (dots per inch) of the printer model. If you choose a density that the printer is incapable of printing, then the density will be set to the nearest printer dot.

Ratio

Sets the wide-to narrow bar ratio. Range (2.0 to 3.0, in .1 increments)
Example: Ratio set to 2.0 means the wide bars of the barcode, are twice as wide as the narrow bars.

Vertical/Horizontal Line Field

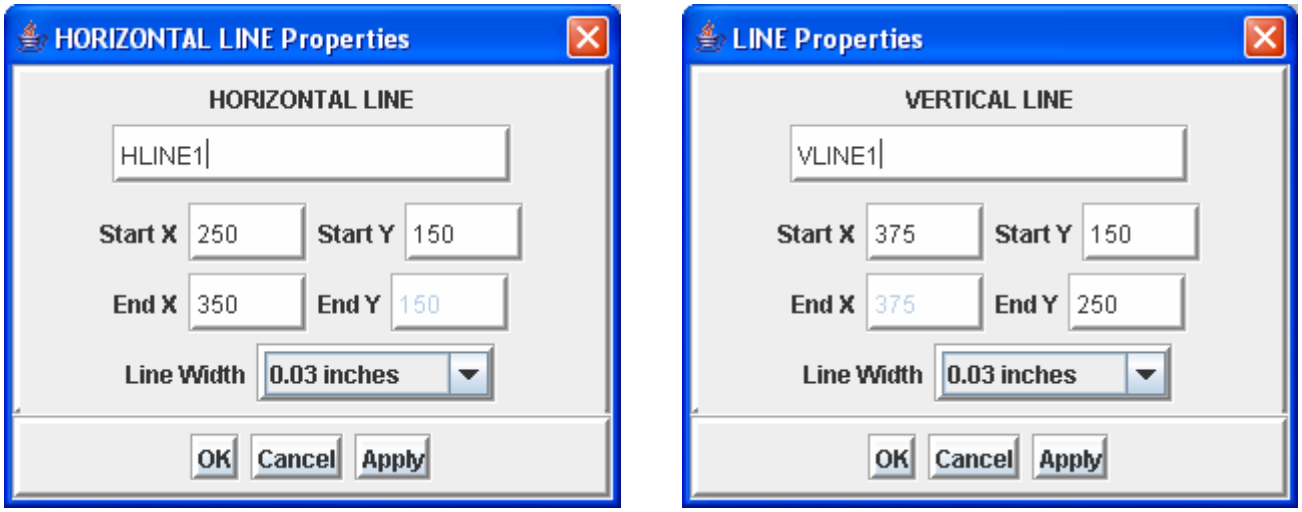
To add a line to the label template: Click on the button and then click-and-drag your cursor either up and down for a vertical line or left to right for a horizontal line. Lines are selected and/or moved in the same manner as other Label-Objects. However, in the event that your format design specification calls for a line thinner than the default [0.03 of an inch] selection may be difficult.

In the event of difficulty selecting a very thin line, follow this procedure:

1) For a horizontal line, place the point of your cursor just above the line before clicking to select. Often the bounding box will respond, if for some reason the line does not.

2) For a vertical line, place the point of your cursor just to the left of the line before clicking to select. Often the bounding box will respond, if for some reason the line does not.

Line Properties



Field Name This is a unique identifier for this particular field.
Example: HLINE1 or VLINE2 Every time the field is added to the label template the number after the field name increases by 1.

Start X/Start Y Displays the starting coordinates of the line field. The units are displayed in 1/100th of an inch (100 equals 1 inch). After the field has been placed on the label template, use this setting to make your fine adjustments.

End X/End Y Displays the ending coordinates of the line field. The units are displayed in 1/100th of an inch (100 equals 1 inch). After the field has been placed on the label template, use this setting to make your fine adjustments.
Note: Horizontal lines only allow you to change the 'End X' coordinate and the Vertical lines only allow you to change the 'End Y' coordinate.

Line Width Sets the thickness of the line.
Example: 0.75 inches will create a line that is ¾ of an inch thick.

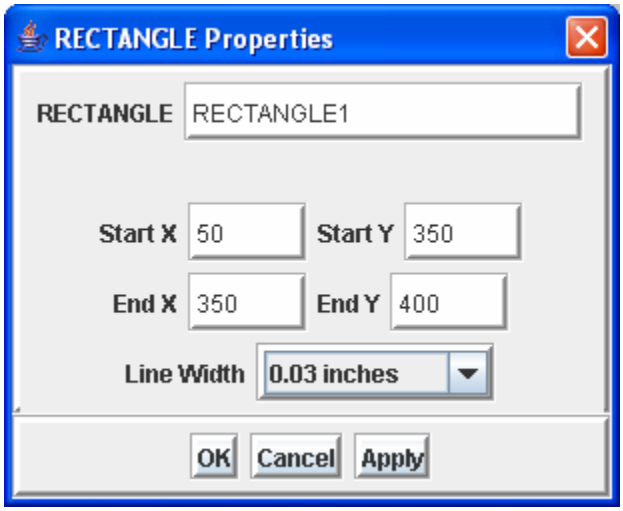
Rectangle Field

To add a rectangle to the label template: Click on the button and then click-and-drag your cursor either to the approximate size of the rectangle you want to create. Rectangles are selected and/or moved in the same manner as other Label-Objects. However, in the event that your format design specification calls for a Rectangle thinner than the default of 0.03 of an inch selection may be difficult.

In the event of difficulty selecting a very thin lined rectangle, follow this procedure:

On a horizontal side, place the point of your cursor just above the line before clicking to select; or on a vertical side, place the point of your cursor just to the left of the line before clicking to select.

Rectangle Properties



Field Name This is a unique identifier for this particular field.
Example: RECTANGLE1 or RECTANGLE2
Every time the field is added to the label template the number after the field name increases by 1.

Start X/Start Y Displays the starting coordinates of the rectangle field. The units are displayed in 1/100th of an inch (100 equals 1 inch). After the field has been placed on the label template, use this setting to make your fine adjustments.

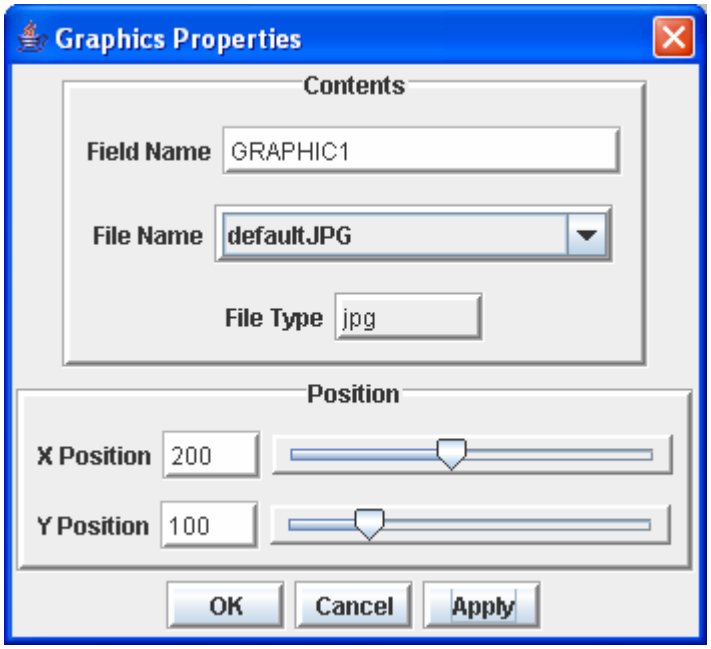
End X/End Y Displays the ending coordinates of the rectangle field. The units are displayed in 1/100th of an inch (100 equals 1 inch). After the field has been placed on the label template, use this setting to make your fine adjustments.

Line Width Sets the line thickness of the rectangle.
Example: 0.75 inches will create a rectangle that has a line thickness of ¾ of an inch.

Graphic Field

To add a graphic to the label template: Click on the button, and then click on the label template where you want to add the graphic. The LDD-Type that corresponds to a GRAPHIC is FILE. In order to place a GRAPHIC on a label format, a FILE object must be created (designated) in the LDD.

Graphic Properties



Field Name This is a unique identifier for this particular field.
Example: GRAPHIC1 or GRAPHIC2 Every time the field is added to the label template the number after the field name increases by 1.

File Name Sets the file to use for this field. The dropdown box will only show the fields in the LDD that have been set up as a 'FILE' field.

Note: When starting a label format from scratch, BARCODE 2000 delivers with sample images with the following names: defaultJPG.jpg or defaultPCX.pcx. They are stored in the images sub-directory The system will automatically place whichever one of these objects is appropriate for your chosen printer model into the current LDD to avoid errors. ALSO, some thermal printers do not support graphic image printing.

File Type Displays the type of file. (PCX, JPG, etc.)

Non-supported formats In the event you are using a "universal" LDD for all your formats, a message appears in the file-type listing to indicate that listed images may not be supported by your particular printer model; simply "fileName:imageType not supported"

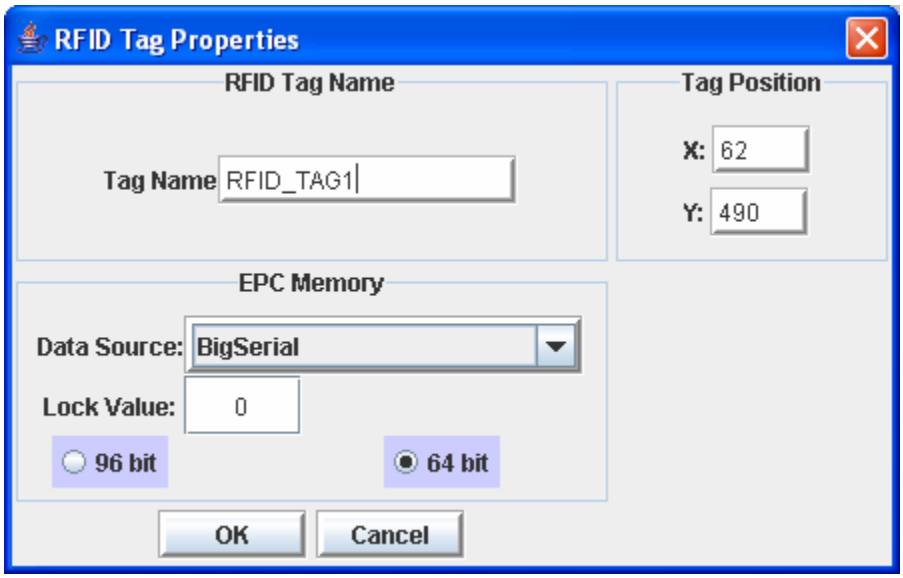
X Position / Y Position Displays the current location of the Graphic field using the lower-left corner as the starting point. The units are displayed in 1/100th of an inch (100 equals 1 inch). After the field has been placed on the label template, use this setting to make your fine adjustments.

RFID Field

To add an RFID tag to the label template: Click on the RFID button, click on either the GEN 1 GEN 2 button, and then click on the label template where you want to add the RFID tag.

1st Generation RFID Tags

RFID Gen1 Properties



Tag Name This is a unique identifier for this particular field. **Example:** RFID1 or RFID_TAG1 Every time the field is added to the label template the number after the field name increases by 1

Tag Position Displays the current location of the Rfid Tag field using the lower-left corner as the starting point. The units are displayed in 1/100th of an inch (100 equals 1 inch). After the field has been placed on the label template, use this setting to make your fine adjustments.

EPC Memory Block

Data Source A list of fields you can use for the barcode field from the LDD data source.

Lock Value Gen1 tags can have a lock value: It is an integer value from 0-255. 0 is not locked

Memory Size Radio Buttons: EPC Memory comes in two sizes: 96 Bit tags & 64 Bit tags

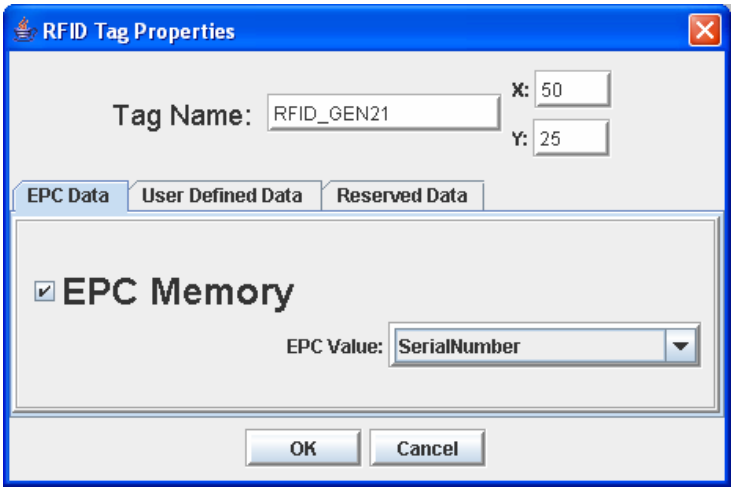
2nd Generation RFID Tags

RFID Gen2 Properties

Tag Name This is a unique identifier for this particular field. **Example:** RFID2 or RFID_GEN21 Every time the field is added to the label template the number after the field name increases by 1

Tag Position Displays the current location of the Rfid Tag field using the lower-left corner as the starting point. The units are displayed in 1/100th of an inch (100 equals 1 inch). After the field has been placed on the label template, use this setting to make your fine adjustments.

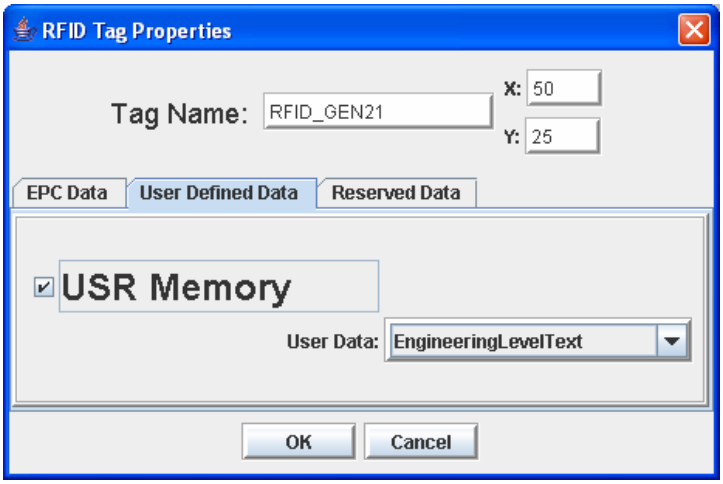
RFID Gen2 – EPC Data Properties



EPC Value This is a 96 bit field that contains EPC Memory data. See tag specifications for usage.

[Data Source] A list of fields you can use for the this field from the LDD data source.

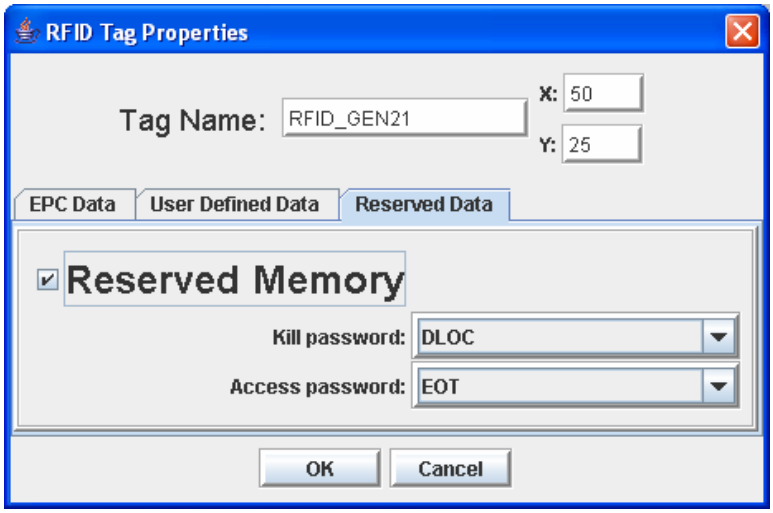
RFID Gen2 – User Defined Properties



User Data This is a 256 bit field that is defined by the user. See tag specifications for usage.

[Data Source] A list of fields you can use for this field from the LDD data source.

RFID Gen2 – Reserved Data Properties



Kill Password This value is a 16 bit field that contains the password to kill the tag. See tag specifications for usage.

[Data Source] A list of fields you can use for the barcode field from the LDD data source

Access Password This is a 16 bit field that contains the password to access the tag. See tag specifications for usage.

[Data Source] A list of fields you can use for the barcode field from the LDD data source

BARCODE 2000 Designer Dialogs by Appearance or Relevance

Label Properties

The Label Properties dialog is where you set the label specifics such as the name of the label, printer model, printer speed and label dimensions.

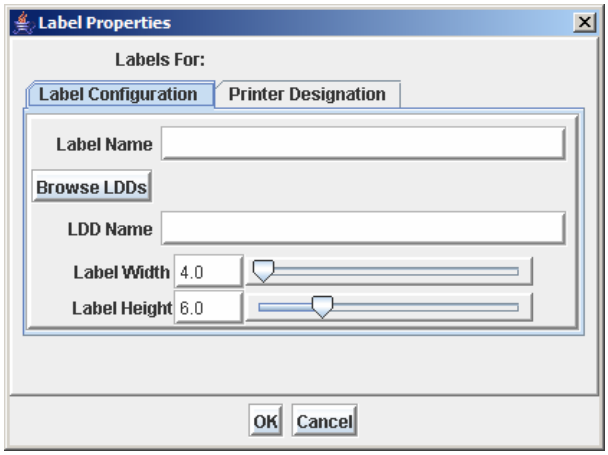


Figure I-1 Label Configuration Screen

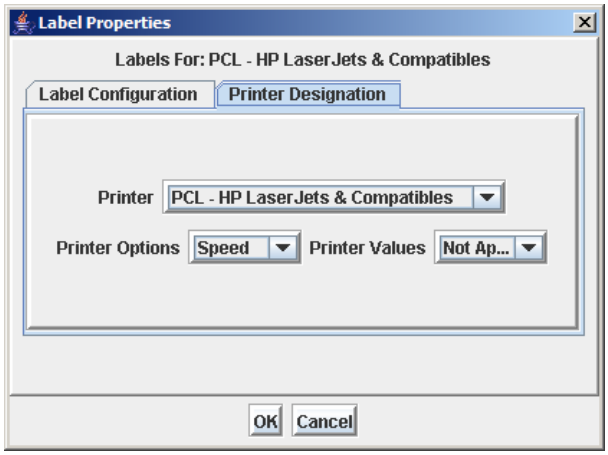


Figure I-2 Printer Designation Screen

Label Configuration Settings

Label Name	Defines the filename for the label template
Browse LDDs	Opens a file chooser dialog to select an LDD for the new label format. If no selection is made, the system
LDD Name	Defines the LDD file that will be used for the label template. *Use the 'Browse' button to select a current LDD file.*
Label Width	Sets the width of the label template in inches. **Example:** 4.0 = 4 inches | Max = 25.0
Label Height	Sets the height of the label template in inches. **Example:** 6.0 = 6 inches | Max = 25.0

Printer Destination and other printer settings

Printer	Sets the printer model for the label template. *Once the printer is chosen for that label template it cannot be changed.*
Printer Options	Set certain printer options for the label template. ***Note:*** *As of the version 7.0 release, only the printer speed options can be set. As more printer options are implemented the printer options list will reflect those additions*
Printer Values	Displays and sets the value(s) corresponding to the Printer Options

Page Setup Dialog

The settings in this area include margins, labels per page and across, LDS default printer and how LDS uses the default data in the LDD.

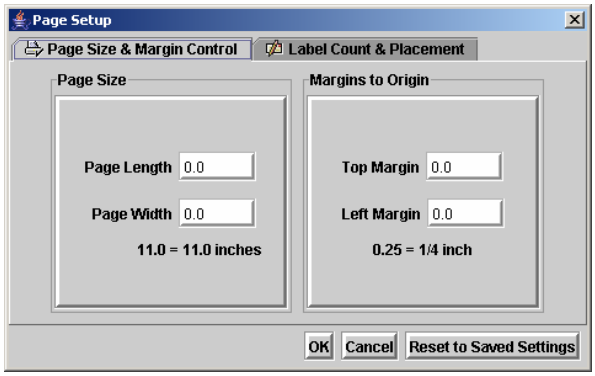


Figure 1-J: Page Size & Margin Control

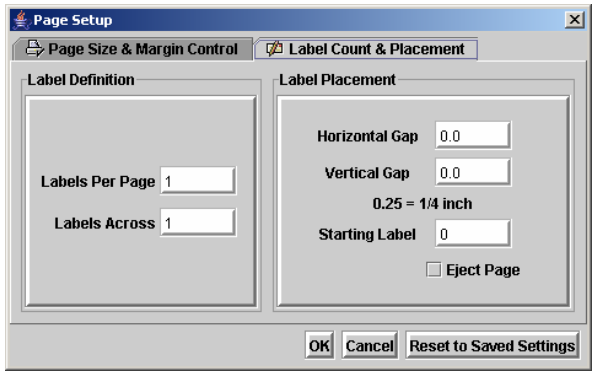


Figure 2-J: Label Count & Placement

Page Size Settings

Page Length — Sets the overall length of the page. This setting is most commonly used when printing to Avery label sheets (address labels, etc.). **Example:** 8.50 = 8½ inches.

Page Width — Sets the overall width of the page. This setting is most commonly used when printing to Avery label sheets (address labels, etc.). **Example:** 11.00 = 11 inches.

Margins to Origin Settings

Top Margin — Sets the top margin for the label template. **Example:** 1.25 = 1¼ inch.

Left Margin — Sets the left margin for the label template. **Example:** 1.50 = 1½ inch.

Label Definition (Count)

Labels Per Page — Sets the number of labels on the label stock. This setting is most commonly used when printing to Avery label sheets (address labels, etc.).

Labels Across — Sets the number of labels across the label stock. This setting is most commonly used when printing to Avery label sheets (address labels, etc.).

Label Placement

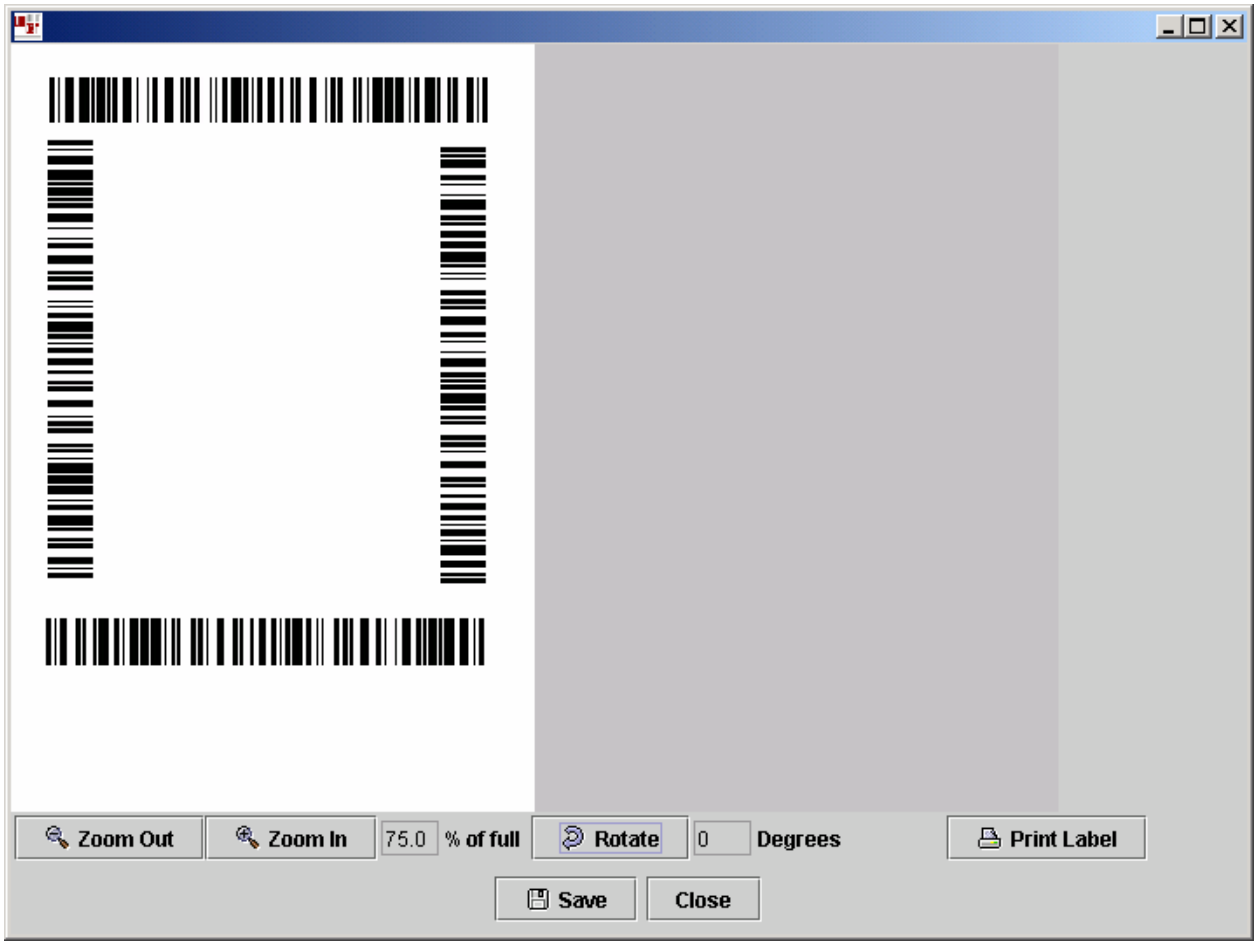
Horizontal Gap — Sets the distance, horizontally, between the individual labels on the label template. This setting is most commonly used when printing to Avery label sheets (address labels, etc.) **Example:** 0.25 = ¼ inch.

Vertical Gap — Sets the distance, vertically, between the individual labels on the label template. This setting is most commonly used when printing to Avery label sheets (address labels, etc.) **Example:** 0.25 = ¼ inch.

Starting Label	Sets the starting label on a page at print time. This setting is most commonly used when printing to Avery label sheets (address labels, etc.). **Example:** If set to 5, then the first 4 labels would not print. This is useful if you have left over labels on a sheet that can be used, instead of discarding the used sheet.
Eject Page	Determines whether or not to print a black page after each label (basically a Form Feed).

Print Preview Dialog

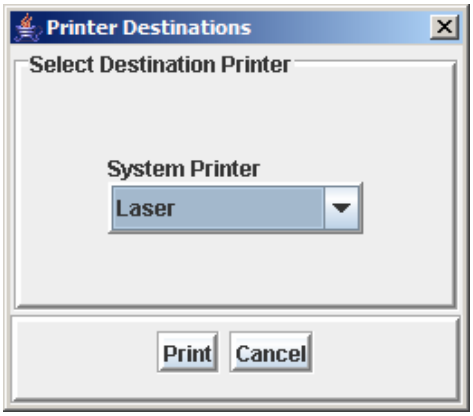
Print Preview is how you can see the label in the correct Rotation. If you need to spin a label for proper placement you can rotate to any of four angles, 0, 90, 180 & 270 degrees.



Zoom Out	Makes the image smaller. The percentage of normal size displays in the **% of full** window.
Zoom In	Makes the image larger. The percentage of normal size displays in the **% of full** window.
Rotate	Moves the image ***& the PRINTED image*** in one of four angles. 0 90 180 & 270 degrees.
Print Label	This will print the label to printer setup in the LDS.
Save	It will save the label in the displayed rotation so all the labels will print with that position.
Close	This will close the label

Printer Destination Dialog

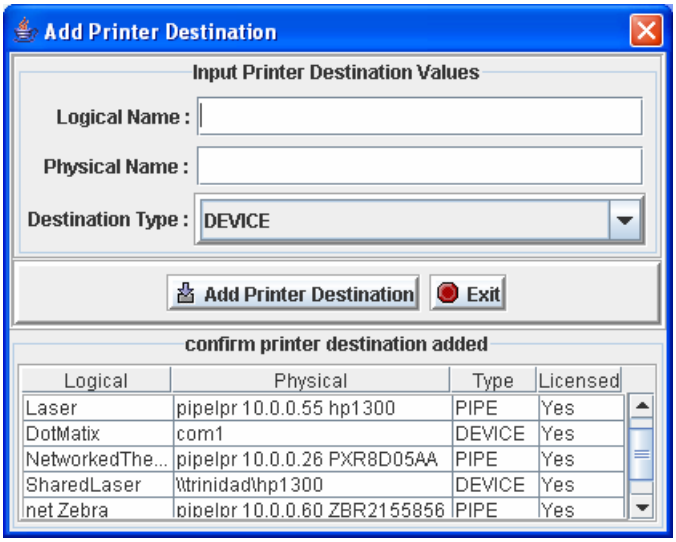
Printer Destination sets the system printer for the ***Print*** events in BarCode2000. The current label template will print to the selected destination. Destinations are set in the Printer Configuration Utility. Your selected printer MUST be set as one of the "installed" drivers in the *printers.lst* file.



Default Printer Sets the default printer LDS will use when printing a test label. This field is populated from entries in the BC2000.cfg file that is setup by using the Configuration component.

Add Print Destination Dialog

This dialog allows the user to add a print destination from within the Designer environment. Any and all print destinations can be added, changed or deleted using the *Configuration utility*.



Logical Name The human recognizable name of the destination printer

Physical Name The system recognizable name of the destination printer – see below.

Destination Type

- DEVICE: a printer connected via network, com port or parallel port
- PIPE: a printer accessible via a server address; uses either ***lpr*** in UNIX or the ***pipelpr*** utility provided with the WIN32 installation package. This method requires the print-server name or address and the internal name of the printer to work properly.
- STDOUT: a printer that takes a print stream directly from the formatter

Add Printer Destination The new print destination shows on the list after this button is pressed.

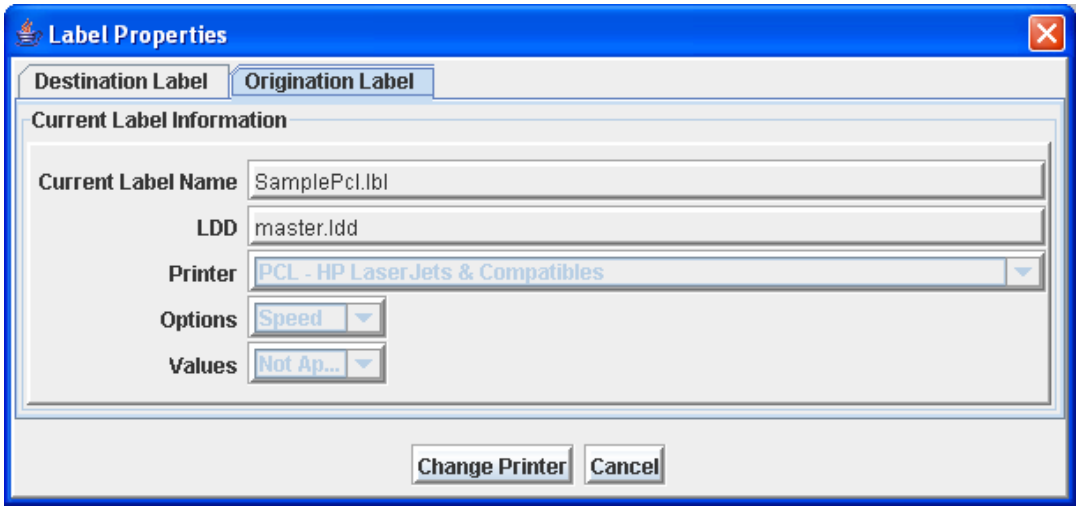
Print Mapping Dialog

This dialog gives the user the ability to "copy" an entire label format to another destination printer. This utility was developed because each driver is different across printer brands.

Fonts are especially tricky with respect to mapping. With this in mind, the mapping utility makes a "best effort" to map an existing format across driver lines. Keep in mind, mapping is not exact. For example, many driver lines only use bitmap fonts. Bitmap fonts are referred to as "Limited" fonts in the designer. By limited, we mean they are only available in certain sizes [point size and characters per inch or CPI] and do not reliably map to scalable or true type fonts. When available, mapping algorithms will try to convert format fonts to a scalable or true type font, but it is not always possible.

Print Mapping Properties

Original Label Properties – Fixed values that are NOT changeable by the user



Current Label Name The name of the originating format

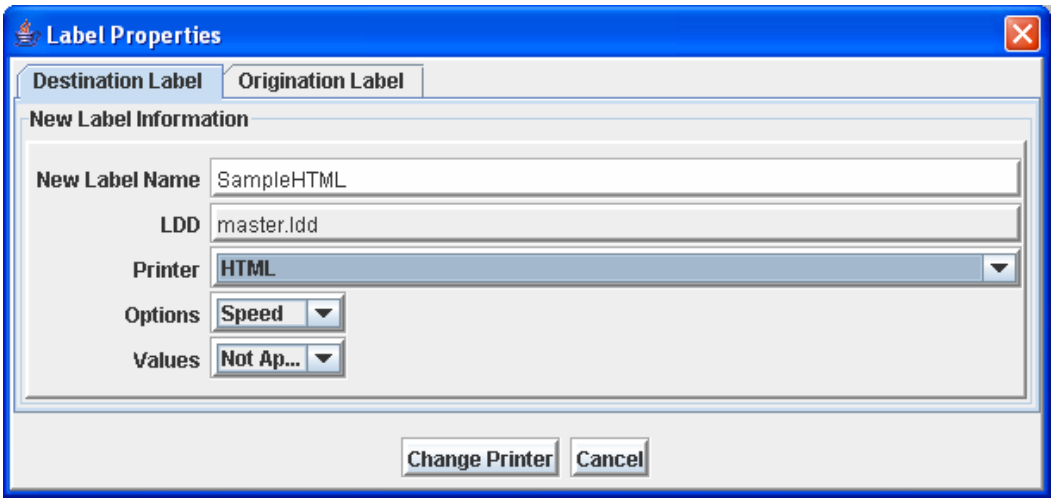
LDD The data source of the originating format

Printer The target printer for the originating format

Options Target printer options

Values Target printer option values

Mapping Destination Information



New Label Name The name of the newly mapped format

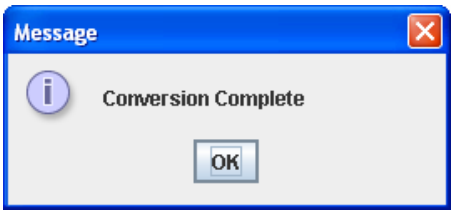
LDD The data source of the originating format –fixed cannot be changed

Printer The target printer for the newly mapped format

Options New printer options

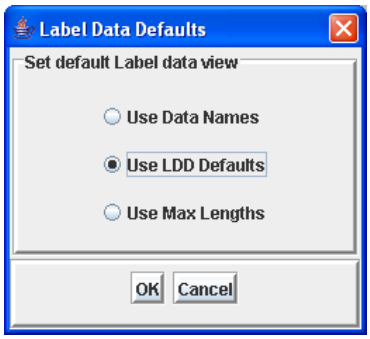
Values New printer option values

Mapping is complete when the following message box appears



Label Data Defaults Dialog

This dialog allows the label designer to choose the form that *DATA* fields are displayed and printed.



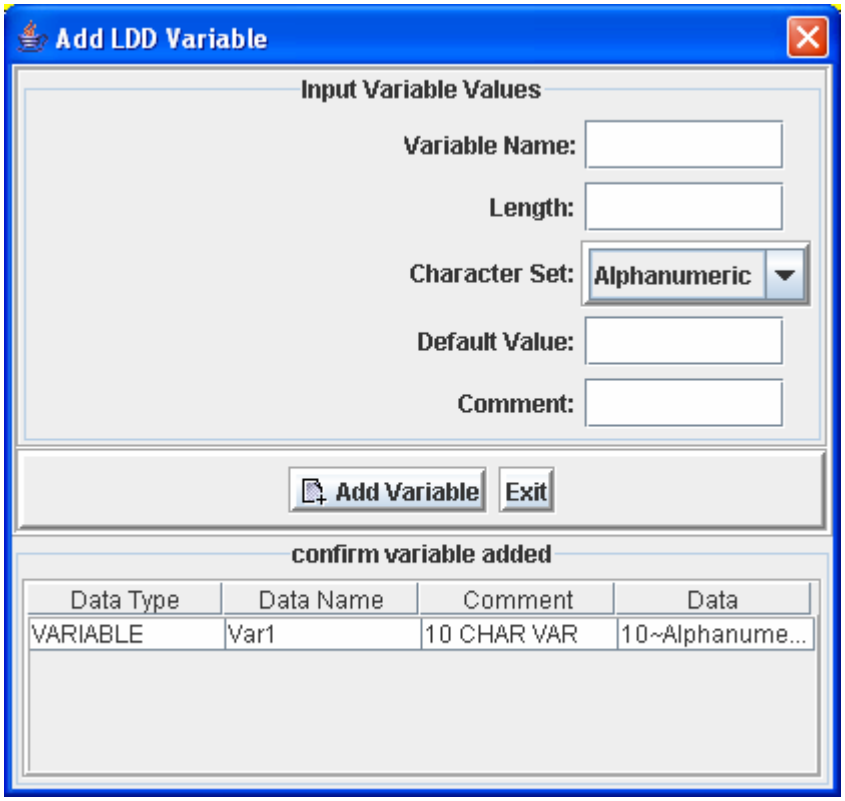
Use LDD Defaults/ Use Maximum Length	This determines what data is printed for the test label when printing from LDS. The 'Use LDD Defaults' setting will use the value specified in the Default box of that specific field in the LDD. The 'Use Maximum Length' setting will use characters ("X") for the data in order to give you a better sense of the maximum length of the field when printed.

Add LDD Entry Dialog

It is possible to add a variable to the data dictionary from within the designer environment. Simply use the menu **[Dictionary→ Add LDD Entry]** or click on the toolbar icon

Add Variable to the LDD Dialog

See the LDD field type Variable for more information



Variable Name You choose the name of your variable

Length This is the max length of the variable

Character set Alphanumeric, Binary, Hexadecimal, Numeric (Decimal)

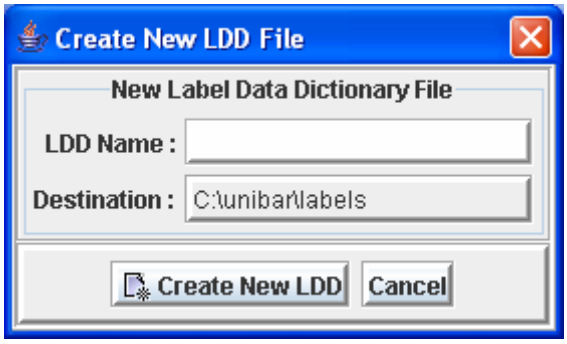
Default Value This is the value seen when the label data view is set to LDD defaults

Comment Leave a note to remember what the variable is meant to represent

Create LDD Dialog

Create a new Label Data Dictionary (empty LDD) from within the Designer to start a new format.

Create New Empty LDD



LDD Name Choose the name of the new Data Dictionary

Destination Currently fixed to the labels sub-directory of the installation directory

Create New LDD Button Press it and you are finished

Introduction to Interactive Data Entry (IDE)

Purpose

IDE is used primarily for printing on-demand labels. Once a label is chosen, all the variable fields are displayed for data entry. Variable fields that have a 'Default' set in the LDD will be displayed as default data on the data entry screen. Also, you can use this program for printing print jobs that have been saved or created.

Menu Bar

The menu bar is located at the top of the IDE main window. This menu bar is similar to other menu bars found in most Windows and Java applications.

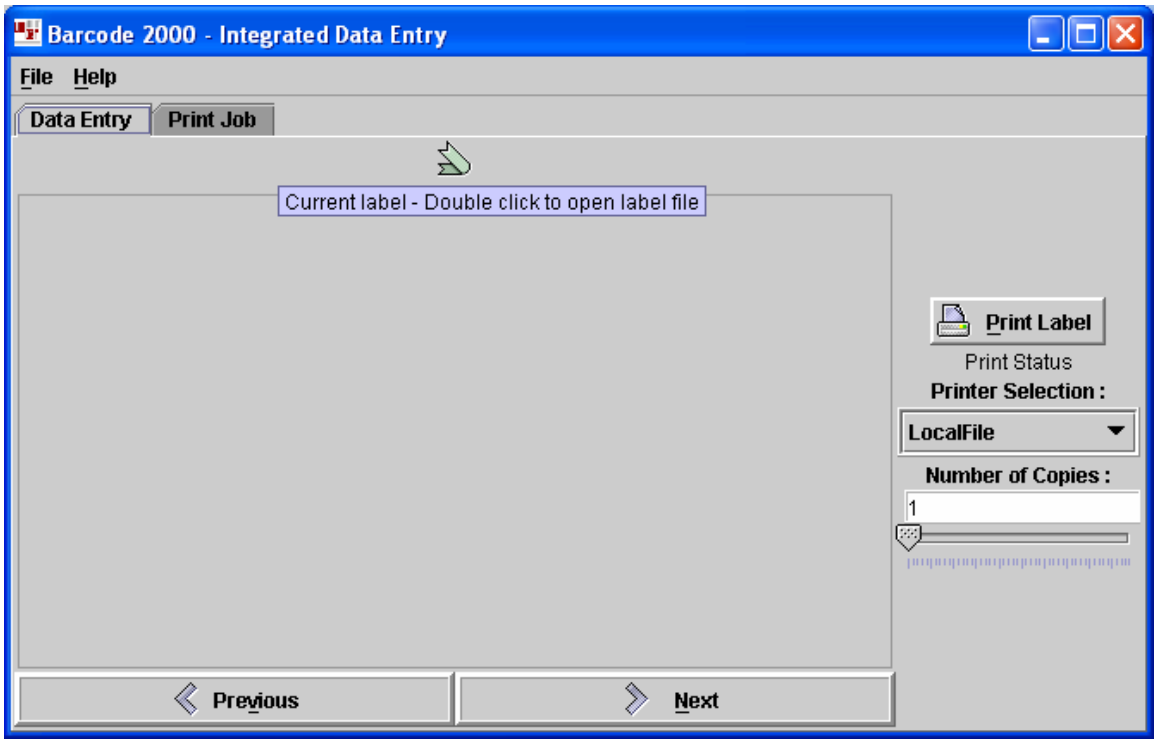


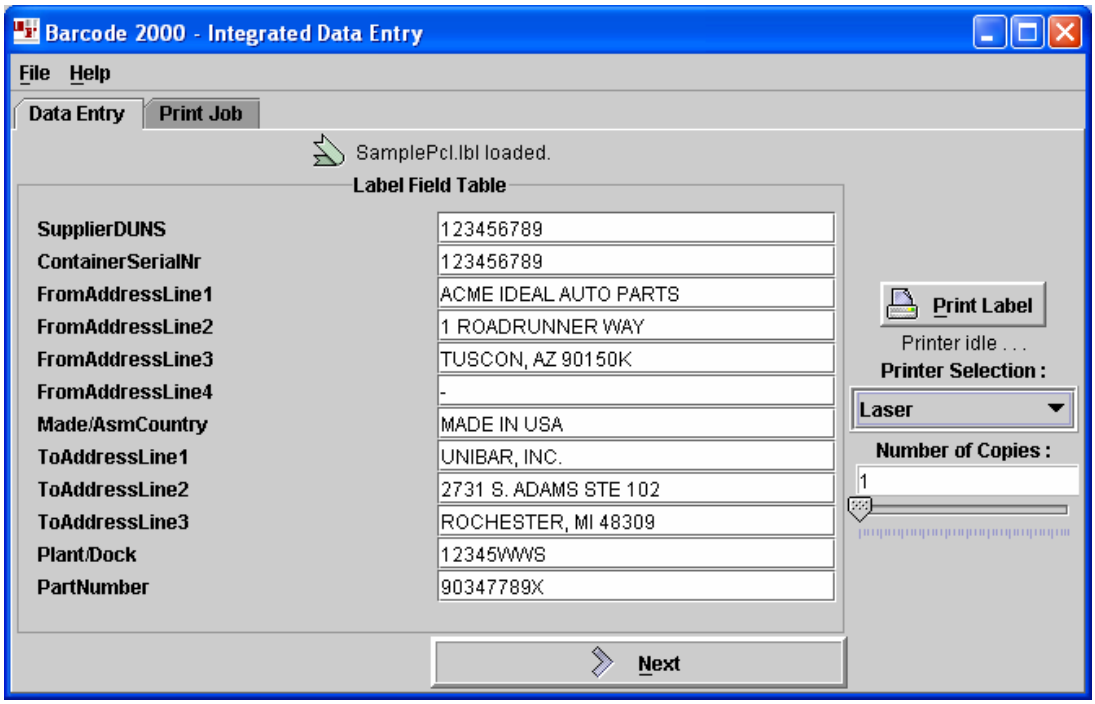
Figure 1-K: IDE Dialog Interface

File To open a label template; quit the application

Help To find help regarding IDE operations; current IDE version number (About)

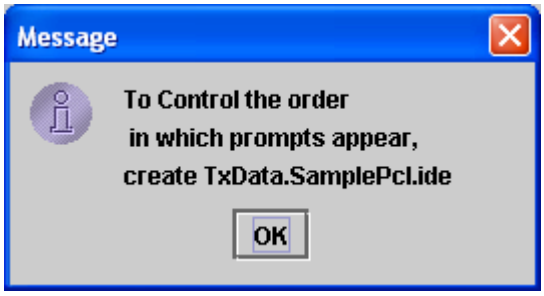
Data Entry

Once the label is chosen, all the variable fields will be displayed in this window. If there is a default value set in the LDD, then the data will be displayed in the textbox adjacent to that field.



Transaction Data Record

When a label format is loaded into the IDE interface, the system searches for a transaction data record. The naming convention for the data record is "TxData." + label name + ".ide". If it does not find the entry you will see this pop-up box. A data record is used in the Print Job utility (see below.)



Once all the necessary data is entered, set the printer by using the dropdown box (see figure I-L). Select the number of copies you want printed by using the slider or by typing the number into the textbox (see figure I-M).

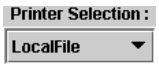


Figure 1-L: IDE Toolbar

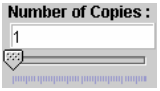
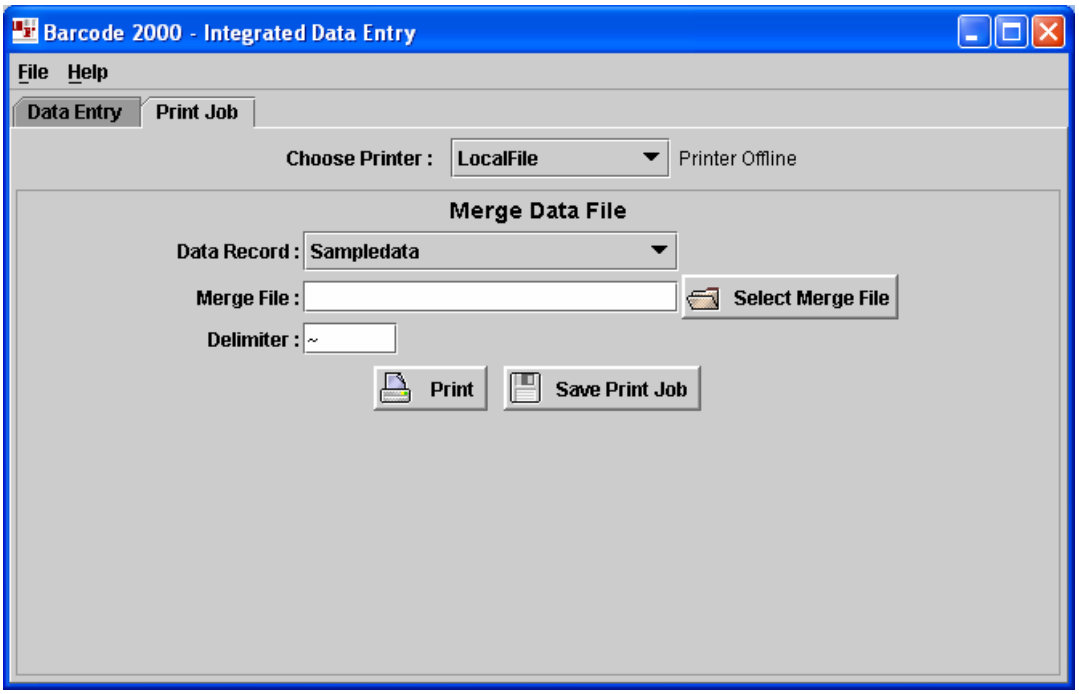


Figure 1-M: IDE

Print Job

To process a data file using IDE, you must setup a TXDATA field in the LDD. Your data file should be in ASCII delimited form. To process the data file, select the label you want to use. Second, select the Data Record. There is a section called Merge Data File, this allows you to

select the TXDATA to use, the location of the ASCII file and the delimiter that is used in the data file. After all the options have been set in the Merge Data File section, print the job by selecting the Print button on the Print Job screen.



Choose Printer — Sets the default printer IDE will use when printing a print job. This field is populated from entries in the BC2000.cfg file that is setup by using the Printer Configuration component.

Data Record — Sets the data record that is defined in the LDD as TxData field. *See the LDD section for more information on the TXDATA field.*

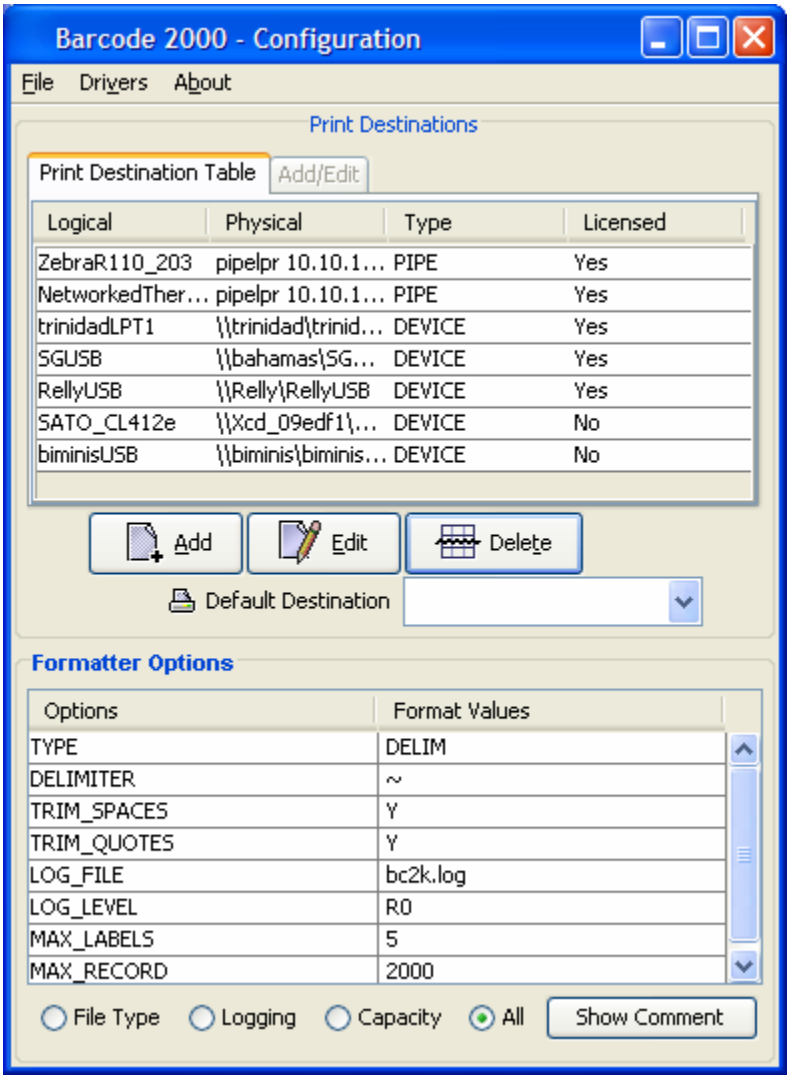
Merge File — Sets the location of the data file to be used with the label template chosen.

Delimiter — Sets the field separator that is being used for the file being merged from the 'Merge File' section.

Introduction to Configuration Tool

Purpose

This operation allows you set the printers, error options, delimiter, and other options used when printing. This tool modifies the BC2000.cfg file, which is located in the Unibar™ base directory.



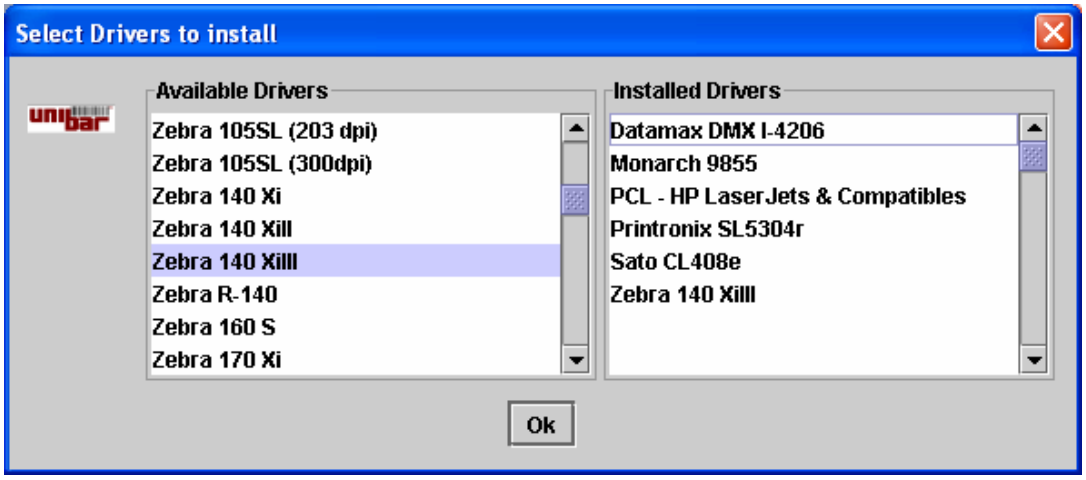
Menu Bar

The menu bar is located at the top of the Configurations main window. This menu bar is similar to other menu bars found in most Windows and Java applications.

Figure 1-N: Configuration Menu Bar

File

- Save the current settings to the BC2000.cfg file
- Exit the Configuration application.



Drivers Displays a list of printer drivers available and installed. Installed drivers will be displayed in the LDS when you create a new label template. To add a printer driver to the 'Installed' section, double-click on the driver name. To remove the printer driver from the 'Installed' section, double-click on the driver name.

About Displays the current Configuration version number.

Print Destination Table – Add/Edit page

This section allows you to define the printers that the Formatter is going to use at print time. Each printer definition consists of a logical name, physical name, the type of output expected and licensing (See Appendix G for more information on the Print Destination Table).

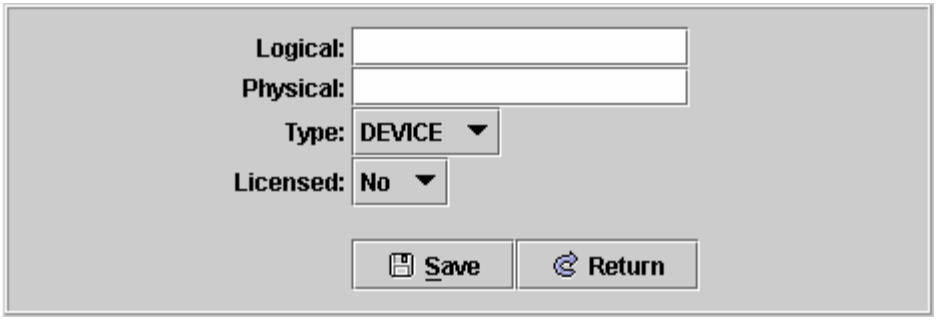


Figure 1-O: Configuration Tool

Logical Sets a name, assigned by the user and is the name by which users refer, to the printer. It can be any useful name, such as ACCOUNTING or DOCK_10. It is a text string of up to 40 characters.
Note: The 'Logical' parameter cannot contain any command line delimiters (space, comma, equal sign, slash, etc.) because it may be used on the command line.

Physical The meaning of the 'Physical' name depends on the destination 'Type'. For DEVICE it is the device name. For PIPE destinations, 'Physical' name is the shell command line that is passed to the system when opening the pipe stream.
Example:
Type = DEVICE Physical = LPT1
Type = PIPE Physical = lp –d Myprinter

Type Describes how the output is handled. The output from the Formatter can be handled one of three ways:
DEVICE - is used for writing directly to I/O devices such as /dev/lp0, /dev/tty1a, etc.
PIPE - is used to specify a pipe command string *(commonly used in Unix systems to pipe the output to the lp print command).*

Licensed Sets the printer definition to 'Yes' or 'No'. If the 'Licensed' is set to 'Yes' then it will be moved to the top of list in alphabetical order.

Note: You are able to have as many printer destinations as you wish, but you can only use the number of license printer destinations that you purchased. The license key determines the number of usable printer destinations.

Default Destination

This specifies the default printer by its 'Logical' name.

Figure 1-P: Default Destination

Formatter Options

This section is for setting options that are available at print time.

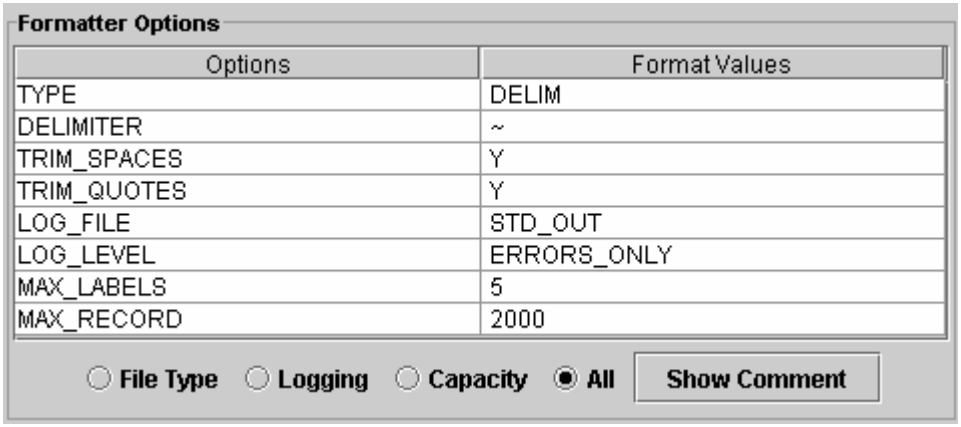


Figure 1-Q: Formatter Options

TYPE Sets the classification of data that will be sent to the Formatter. There are only 2 possibilities, delimited or fixed. Delimited data has some sort of field separator (field1~field2~field3). In fixed data, the data is in a fixed position and never changes from record to record (field1field2field3).

DELIMITER Sets the delimiter used in the data if the 'TYPE' is set to delimited. The delimiter should be a unique character/symbol. Unibar™ recommends using a '~' for a delimiter. Other possible delimiters could be a tab, !, @, etc.

TRIM_SPACES This setting is either 'Y' or 'N'. If set to 'Y', then the Formatter will remove the leading and trailing spaces that are in a field.
Example:
Data = {space}{space}001234
Print = 001234 (notice no spaces in the front of the data)

TRIM_QUOTES This setting is either 'Y' or 'N'. If set to 'Y', then the Formatter will remove the leading and trailing quotes (") for that field.
Example:
Data = "Field 1","Field 2","Field 3"
Converted to: Field 1,Field 2,Field 3

LOG_FILE This determines how the Formatter reports errors encountered. For example if the 'LOG_FILE' is set to STD_OUT, the all errors will be reported to standard out. This setting could be a filename where errors are written to.

LOG_LEVEL Sets the error log level when the Formatter is running.
Note: Should be set to 'ERRORS_ONLY' unless instructed by Unibar's ™ technical support staff.

MAX_LABELS Sets the number of different label templates that will be loaded in a printer memory at one time. For example, if set to 5, then on the 6^{th} distinct label type encountered during a single print job, the constant part of the labels and graphics will be deleted from the printer memory.
Note: If the printer does not support storing labels in the printer memory then this setting will be ignored.

MAX_RECORD Sets the maximum length of a data record being sent to the Label Formatter. Records longer than this will have truncated data. If the truncated data needed data incomplete labels may be printed.

Chapter 2 - Label Formatter - Direct Use

The Label Formatter is a program that takes an input file or standard in (stdin) and prints a batch of labels. In general, there is one input record for each label to be printed (multiple copies may be specified). The input record contains data that varies from label to label, such as the shipping address or lot number, whereas the label template contains data that remains constant, such as the return address or the company logo. The first record contains the data names (defined in the Label Data Dictionary (LDD)) in the same order as in the following data records. An alternative is to use a Transaction Data Definition String, which is created in the LDD.

The label template produced by the Label Designer with field definitions form the Label Data Dictionary (LDD) tells the Formatter how to format each label. The Formatter ***(ubfmt)*** takes its input from *stdio,* so it will accept input either from a file (" ubfmt <input_file> ") or piped from an application (" user_program | ubfmt "). See Examples of program code for 'C' and Java languages.

Example of using the Formatter from a command line

ubfmt /L=SampleBarcodes /C=1 /P=LocalFile /DRtxdata

48309~48309~0123456789~

**Note - TXDATA is defined in the LDD*

Example of using the Formatter by piping the data (In C)

```
#include <stdio.h>
#Include <stdlib.h>

/* fmt_pipe.c  - Sample Unibar™ Formatter pipe interface program
   assumes that BARCODE 2000 is installed in c:\Unibar (see _popen call below)
*/

void main( void )
{

   FILE   *fmt_pipe;     /* Formatter pipe I/O structure pointer */

   /* Execute Formatter using _popen command: returns handle to the input pipe */

   /* _popen(char *command, char *mode)               */

   if( (fmt_pipe = _popen("c:\\Unibar\\ubfmt.exe /L=Sample.lbl /p=LocalFile",
"w")) == NULL)
         exit(1);

   /* write data to Formatter through the standard in pipe
      using fprint command */

   fprintf(fmt_pipe, " /C=1");  /* One Copy */
   fprintf(fmt_pipe, " /DR=Customer Zip+Ship Zip+Purchase Order\n");
      /* Specify Data Record */
   fprintf(fmt_pipe, "48309~48309~0123456789\n");
      /* Specify Record Data */
```

```
   /* Close pipe and print return value from the Formatter */

   printf( "\nFormatter return code: %d\n", _pclose( fmt_pipe ) );

}
```

Example of using the Formatter by passing a file (In Java)

This java application shows a method of execution of the Unibar™ Formatter. The sample label, "Sample", is used and will generate output in the file "OUTPUT.FIL".

To make this application run correctly on your system, changes to the RT. exec call in the main () method below may be necessary. Make sure that the directories in the exec call correspond to the directories of your particular installation of BARCODE 2000.

```
import java.awt.*;

public class fmt_exec
{
      public fmt_exec()
      {     // no action
      }

      public fmt_exec(String title)
      {     // no action
      }

      public synchronized void show()
      {     // no action
      }

      static public void main(String args[])
      {
        new fmt_exec();
        Process Print_Process;
         // Retrieve Runtime environment
        Runtime RT = Runtime.getRuntime();
        try{

        // Execute Formatter using runtime exec method
         // will return a system process
            Print_Process = RT.exec("c:\\Unibar\\ubfmt.exe
            /f=c:\\Unibar\\fmtin.txt /L=SampleBarcodes /p=LocalFile
            /w=c:\\Unibar\\error.out");

        // Loop until exitValue succeeds
        boolean Done = false;
        while(Done == false)
        {
            try{
            int exit;
            // retrieve Print_Process exit code
            exit = Print_Process.exitValue();

            Done = true;
              // exitValue succeeded if we reach this point

            if(exit >= 0)
                System.out.println("Printing finished\n");
```

```
            else
                System.out.println("Printing error\n");
            }
            catch(java.lang.IllegalThreadStateException e)
            {     // exitValue fail
            }
        }
        }catch(java.io.IOException e)
        {         // exec call fail
            System.out.println("IO Exception caught:" + e.getMessage());
        }
    }
}
```

Steps in Creating a Job with Formatter

The Formatter works from a single command line. It may sound simple in that you only have one line to input, but it must be entered in a very specific way, so that the program recognizes each part of the command.

$ ubfmt < datafile.asc /P=logical_printer_name /L=label_name /DRtxdata_name

*Note - TXDATA is defined in the LDD

This example reads as follows:

Call the formatter by name, ***ubfmt***, tell it to take as input the data from ***datafile.asc***, print it to ***logical_printer_name*** (which is defined in the BC2000.cfg), using ***label_name*** as the label format and the Label Data Dictionary entry ***txdata_name*** contains the mapping from the variables to the data in the given data file.

Parameter Listing

The Formatter takes several parameters, some are required and some are optional.

There are four places where parameters can be set. Parameters that are fairly consistent within one site, such as the default printer, can be set in the configuration file. Most parameters can be set on the command line. A group of command line options can be collected into a job file, and then the job file can be specified as a single command line option. Finally, some options can be set in the data input file itself.

When a new parameter setting is encountered, it overrides any previous setting. So a job file overrides the configuration file, the command line overrides the job file and the data file overrides the command line.

*** Refer to table on the following page. ***

The following table summarizes the Formatter parameters:

Parameter	Required	BC2000.cfg	Job File	Command Line	Data File
Copies			Y	Y	Y
Data Record Format (Field Names)	Y				Y
Delimiter		Y	Y	Y	
End					Y
File (input)				Y	
File Type	Y	Y	Y	Y	
Job Field nn			Y	Y	Y
Job File				Y	
Label Name	Y		Y	Y	Y
New Page					Y
New Row					Y
Print Destination	Y	Y	Y	Y	Y

Copies

The syntax is "*/C=n*". 'n' copies of each record are printed. The number of copies accepted is 1 - 999999. This can be set once for the entire print job, or can be set on each label. Once it is set, that number of copies is in effect until it is set again.

Data Record Format

The Data Record Format defines the order and identity of data fields as they will appear in the data stream or file. The data record format is specified in the **LDD** under TXDATA, or it can be located at the beginning of the data file.

`ubfmt /L=label /P=printer /DR`*`txdata_name`*

Where *`txdata_name`* is the name of the TXDATA entry in the LDD for the given label format .

Delimiter

The syntax is "*/DEL= d*", where'd' is the field delimiter. For example, "*/DEL=,*" to make the delimiter the comma character. For delimited files, the field delimiter defaults to the tilde (~) or the default set in the BARCODE 2000 Configuration file but can be overridden with the /DEL parameter. The Formatter takes one character immediately following the equal sign.

End

/End

This signifies the end of the Data Record options and the start of the actual data. There are two methods of specifying the start of data:

- One or more blanks followed by any character except a slash '/'. This character becomes the first data character.
- "*/END*" followed immediately by the first data character, which may be a blank or a slash.

File Input

/F =data file

This command is an alternate to using stdin. When included on the command line, the records in the specified data file are passed to the formatter, enabling you to print a batch of labels. In this file you will include the information that will vary from label to label.

File Type

The syntax is "*/T=Type of File* ". The Formatter supports both delimited and fixed length ASCII files. The /T parameter selects a file type as follows:

/T=D for delimited file type (default)
/T=F for fixed length files

The file type can be also set up in BC2000 Configurator.

Job Field

The syntax is "*/Jnn=<value>* ".

This parameter is used to set a value for a job field. A Job Field is a field on a label whose value may vary from job to job but is constant within a job. These are typically defined in a job file but can be specified on the command line as well. The value cannot contain any command line delimiters, such as a carriage return, space, comma, slash, etc. ('nn' is the number of the job field. For example, " /J01=Plant_ABC " sets job field 01 to the string "Plant_ABC").

Job File

/J=Job File

The specified file is treated as an extension of the command line. Job files provide a way to store a group of parameters that are associated with a certain print job or group of print jobs. In a job file, the pound sign '#' character is used as a comment delimiter.

Label Name

/L= Label Name

This parameter is used to assign a label template for a print job. The Label Name is the name assigned to a label by the user in the Label Designer program. The program looks up the Label Name in the labels.lst file in the "Unibar" base directory. If it cannot be found, the command is

ignored. The Label Name is case sensitive. Label Name can be set on the command line or in a job file. If no label is assigned, the Formatter will abort the print job.

New Page

/NP

Used when a print page is not full and you want to advance to the next page. The *New Page* option is used to signify that this label should begin a new page. Normally this command is used with laser printers with multiple labels per page.

New Row

/NR

Needed only if row is not filled. The *New Row* option is used to signify that this label should begin a new row. This can be used, for example, on laser printers where the label stock has four labels across on each row.

Print Destination

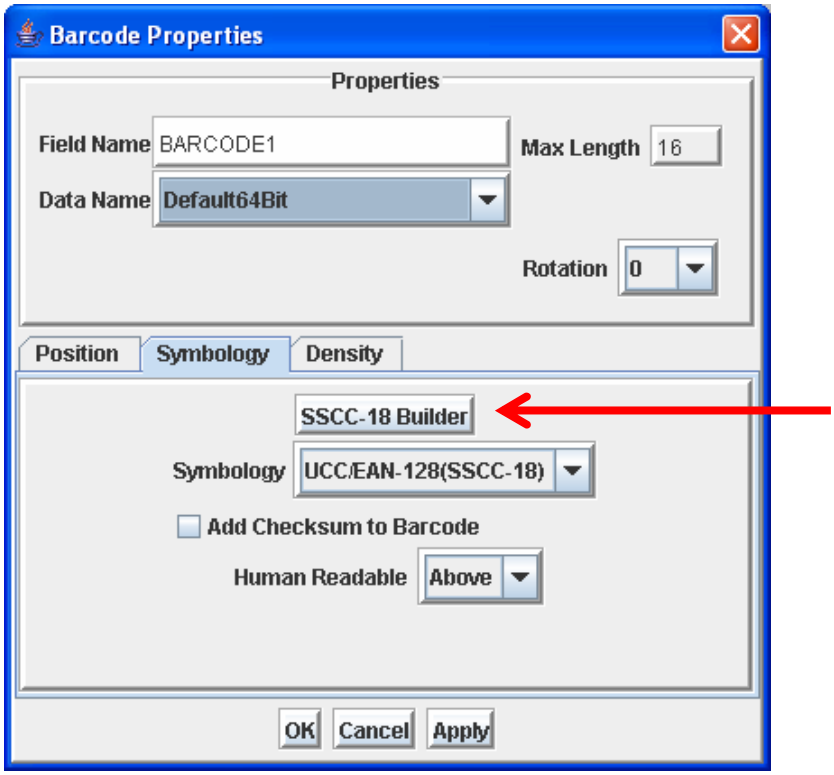
/P=Logical Printer

The Print Destination is the Logical name of a print destination defined in the configuration file BC2000.cfg. If the Print Destination cannot be found in BC2000.cfg, the print job is aborted.

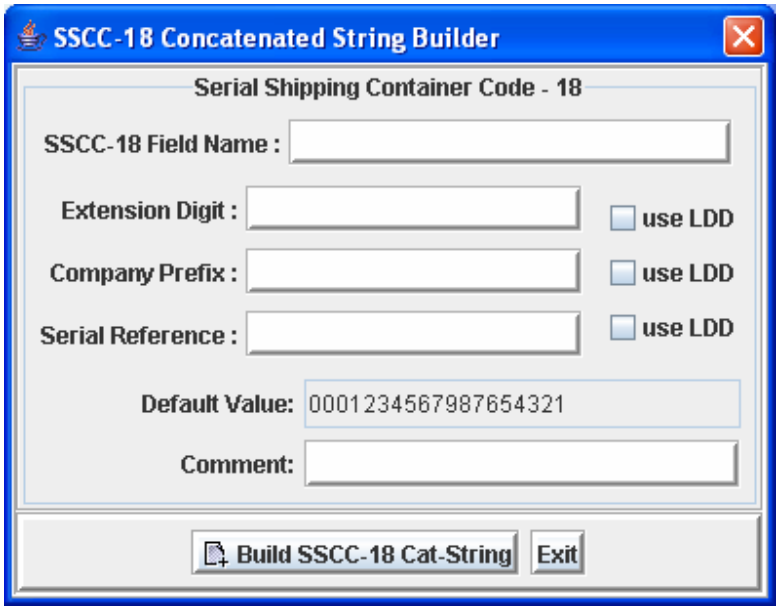
Chapter 3 – Specialty Barcode Symbologies

Serial Shipping Container Code (SSCC)

This is the SSCC from Uniform Code Council, Inc. used for EAN.UCC Systems. To create one, select the Barcode label object and then change the Symbology to *UCC/EAN-128 (SSCC-18).* Doing so will cause the *SSCC-18 Builder* button to appear at the top of the Symbology Tab [see arrow.]



Pressing the button brings up the **SSCC-18 Concatenated String Builder** Dialog. This dialog allows for easy creation of LDD-variable concatenations used specifically for SSCC-18 barcodes.



The fields are as follows:

SSCC-18 Field Name: This is the name in the LDD file.

Extension Digit: Used for increase of Serial Reference – typically only one digit.

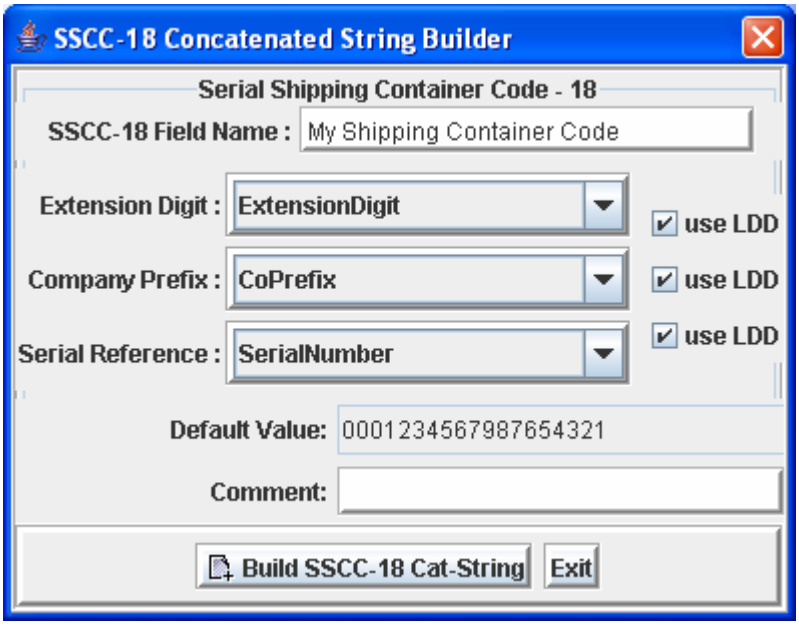
Company Prefix: The EAN.UCC number assigned to your company

Serial Reference: Unique number to identify logistic unit

The Company Prefix plus the Serial Reference add up to 16 digits

Automatic - Check Digit: Calculated one digit number to ensure data integrity

Use LDD: By checking this box, the text entry area becomes a pull-down-box populated by the current label data dictionary entries. Using **LDD** variables when building this field allows for interchangeable variable data on any of the three "aliased" input fields. [see below]

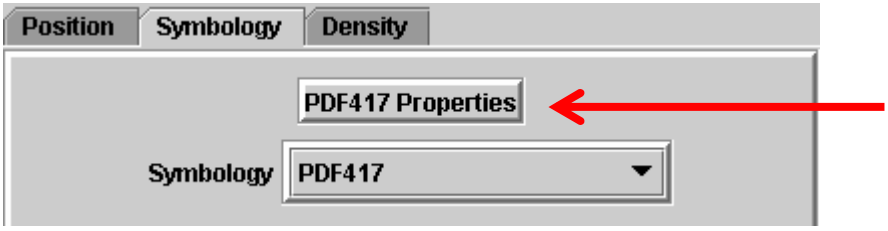


Using this "wizard" to create an SSCC-18 Barcode produces an image on the canvas that should look like the following example:

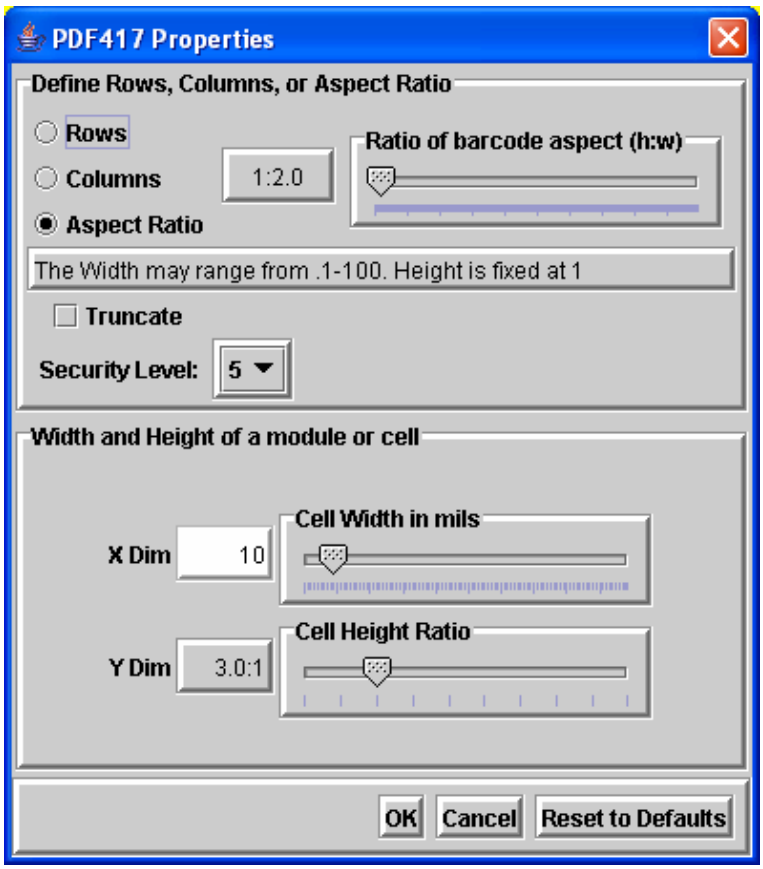
2D – Two Dimensional Barcodes (PDF417)

Barcode 2000 supports the creation of the Two Dimensional Barcode Symbology known as PDF417. This Symbology can hold vast amounts of data. In fact, PDF417 barcodes are sometimes used to hold a redundancy record of a label-formats entire content.

When PDF417 is selected from the Symbology list on the Barcode Properties' Symbology tab, a button appears over the drop-down list.



PDF417 Properties:



Define Rows: The number of rows of cells. Row value range from 3-90; Default of 0 is Auto-sizing

Define Columns: The number of columns of code words. Column value may be 1-30, Default 0

Aspect Ratio: The ratio of barcode aspect. (h:w) The Width ranges from 0.1-100. Height is fixed at 1

Truncate: Selecting this option removes the right edge of the barcode.

Security: Increasing the Security, increases the redundancy of the data in the barcode.

Width and Height of a Module or Cell:

X Dim: Width of a module or cell in Mils. Range is 1 – 100; Default is 10.

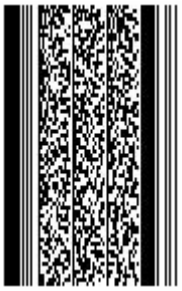
Y Dim: Height of as cell or module; specified as a scalar to the X - dimension

Here are a few examples of what some of the settings may do to a PDF417 barcode.

Aspect 1:2, x=30 (default is 10) basically this makes the barcode uniformly larger.

Y=1.0:1 (default is 3.0:1) Notice how the cells are “squares?”

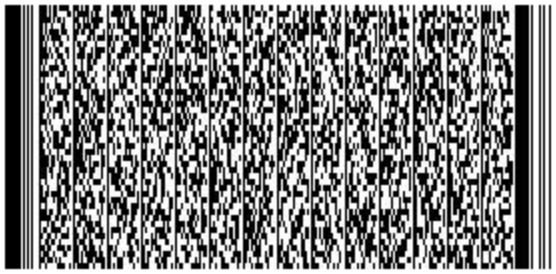
When the aspect width is less than 1, the 2d barcode becomes “tall”



Truncated means the right edge is removed

Defaults with security level “1” the lowest level of redundancy. (default security is level 5)

The same data and settings with security level “8”. The reason for the redundancy is to protect the data if part of the barcode is lost or illegible.



Appendix A – Printer Support

This list is updated continually. In order to see an updated list of printers that Unibar™ supports see our web page at http://www.unibar.com

A

Accumark 3000
Accu-Sort Applicator
Accu-Sort FD-4630
Accu-Sort PTP/Applicator
Action Packaging GS-90
Antonson Range Boss
Antonson Top Hand
Appollo 1
Argox X-1000+
Argox X-2000+
Argox X-3000+
ASI
Astro-Med - LC8000
Astro-Med - Legacy IV
Astro-Med - Range Boss
Astro-Med - Top Hand
Astro-Med - Top Hand 2
AT Info "Autoprint"
AT Info XT500
Atech - 205
Autolabel - 500
Autolabel - XT-500
Avery - ALS 94
Avery - ALX 910
Avery - Control - TLP-40
Avery - Control - TLP-46
Avery - TTX1050
Avery - TTX600
Avery - TTX650
Avery - TTX674
Avery - TTX675
Avery - TTX706
Avery - TTX736
Avery - TTX807
Avery - TTX950
Avery Dennison - 3000
Avery Dennison - 4200
Avery Dennison - 8400
Avery Dennison - 908
Avery Dennison - 9500
Avery Dennison - Datapply 900
Avery Dennison - Datapply 989
Avery Dennison - Datapply 990
Avery Dennison - DCP 5207
Avery Dennison - Intacs 3000
Avery Dennison - Intacs 3520
Avery Dennison - Intacs 3528
Avery Dennison - Intacs 4200
Avery Dennison - Intacs 5000
Avery Dennison - Intacs 5350
Avery Dennison - SPX30
Avery Dennison - TTX450

B

Best Label Systems Versprint
Bluhm Systems Leg Allegro
Bluhm Systems Legitronic 76
Brady 1024
Brady 2461
Brady 3000
Brady 300X
Brady 3481
Brady 360X
Brady 6441
Burford - TTP II

C

C.ITOH - 400
C.ITOH - S4
C.ITOH - T4
C.ITOH - T4300
C.ITOH - T4R
C.ITOH - T4S
C.ITOH - T4ST
C.ITOH - Terminator
C.ITOH - Z4
C.ITOH T4 Terminator
C.ITOH T4 Terminator - Tear Off
cab Hermes 2
cab Hermes 4
cab Hermes 5
Cognitive Code Courier
Columbia - CPA 125
Columbia - CPA 150
Columbia - CPA 250
Columbia - TR 10
Columbia - TR 22
Computype Sigma Plus
Control Print - TLP46

D

Datamax Allegro
Datamax Allegro - Tear Off
Datamax Allegro 2
Datamax Allegro 2 - Tear Off
Datamax Destiny 430
Datamax DMX 400
Datamax DMX 400 - Tear Off
Datamax DMX 430
Datamax DMX 430 - Tear Off
Datamax DMX 600
Datamax DMX 600 - Tear Off
Datamax DMX 800
Datamax DMX 800 - Tear Off
Datamax DMX E-4203
Datamax DMX E-4204
Datamax DMX Ex2
Datamax DMX I-4206
Datamax DMX I-4206 - Tear Off
Datamax DMX I-4208
Datamax DMX I-4210
Datamax DMX I-4212
Datamax DMX I-4308
Datamax DMX I-4406
Datamax DMX I-4604
Datamax DMX W-6208
Datamax DMX W-6308
Datamax DMX W-8306
Datamax Ovation
Datamax Ovation - Tear Off
Datamax Ovation 2
Datamax Ovation 2 - Tear Off
Datamax PE-42
Datamax PE-43
Datamax Prodigy
Datamax Prodigy - Tear Off
Datamax Prodigy Max - Tear Off
Datamax Prodigy Max-203
Datamax Prodigy Max-300
Datamax Prodigy Plus
Datamax Prodigy Plus - Tear Off
Datamax Titan 6200
Datamax Titan 6200 - Tear Off
Datamax XL
Datamax XL - Tear Off
Dennison - 2300

Dennison - 362
Dennison - 4500
Dennison - 5300
Dennison - 5301
Dennison - Accumark 3000
Dennison - Intacs 3000
Dennison - Intacs 3520
Dennison - Intacs 3528
Dennison - Intacs 4200
Dennison - Intacs 5000
Dennison - Intacs 5350
DH Tech 2420E
DH Tech 4220
DH Tech 4220E
DH Tech 8400
DH Tech DP2020
DH Tech DP7000
DH Tech TT2420
DH Tech Upwriter
Diagraph - 8400
Diagraph - Allegro
Diagraph - LPT1040
Diagraph - PA2000
Diagraph - Plus
Diagraph - Prodigy
DPC 5207

E

Eltron 2443
Eltron 2622
Eltron 2642
Eltron 2684
Eltron 2722
Eltron 2742
Eltron 2746
Eltron 2824
Eltron 2844
Eltron 3642
Eltron 3742
Eltron Companion Plus
Eltron Ht-146
Eltron LP/TLP Eclipse
Eltron Orion
Eltron QB440
Eltron QB440V
Eltron QB450
Eltron QB450V
Eltron QB485
Eltron Stratta
Eltron TLP 2046
Eltron TLP 2242
Eltron TLP 2622
Eltron TLP 2642
Eltron TLP 2722
Eltron TLP 2742
Eltron TLP 2824
Eltron TLP 3642
Eltron TLP 3742
Eltron TransPort
Esselte Meto - 8500
Esselte Meto - AS-40
Esselte Meto - MP-420
Esselte Meto - V40
Esselte Meto - V65
Esselte Meto - V85
Esselte Meto Bandit
Esselte Meto Titan Model MP-420
Etigraph - T119
Etimark - MP500T

F

Facit - D450
Facit - D470
Facit - D475
Facit - D680
Facit - T-500
Fargo
Formula 245
Formula 245T
Formula 246
Formula 246T
Freedom T-112
Fuji 905

G

Genicom - 1095
Genicom - 6241
Genicom - 6342
Genicom - 6441
Geo Packaging - G115
Graphic Tech Duraprinter-R

H

HTML
HP LaserJets – See PCL

I

IBM Infoprint 4400 [see list below]
- Model 004 (203 dpi)
- Model 004 (300 dpi)
- Model 006 (203 dpi)
- Model 006 (300 dpi)
- Model 008 (203 dpi)
- Model 008 (300 dpi)

Image – [see list below]
- JPEG 100 dpi
- JPEG 300 dpi
- PNG 100 dpi
- PNG 300 dpi

Intacs - 5000
Intacs - 5500
Intellibar
Intermec - 3100
Intermec - 4000
Intermec - 4100
Intermec - 4630
Intermec - 4830
Intermec - 7421
Intermec - 7422
Intermec - 91
Intermec - C4
Intermec - C407
Intermec - Easycoder 401
Intermec - Easycoder 501
Intermec - Easycoder 501XP
Intermec - Easycoder 601
Intermec - Easycoder 601XP
Intermec 3100
Intermec 3240
Intermec 3400A
Intermec 3400B
Intermec 3400C
Intermec 3440
Intermec 3600
Intermec 4000
Intermec 4100
Intermec 4400
Intermec 4630
Intermec 4830
Intermec 7421
Intermec EasyCoder C4
Intermec EasyCoder F4
Intermec Intellitag PM4i

J

JPG, JPEG,JPE, - see Image

K

Kroy - K4200
Kroy - K4300
Kubota - MP-11

L

Label Jet - 2600
Labelaire - 138
Labelaire - 21
Labelaire - 2135
Labelaire - 2136
Labelaire - 2137
Labelaire - 2137-P
Labelaire - 2138
Labelaire - 3138PA
Labelaire - 362
Lord Label - 96TR

Lord Label - LPA
Lord Label - TR
Lowry - 4121

M

Mars Meo - 1600
Mars Meo - 1632
Mars Meo - 1634
Matthews - 4100
Matthews - 4101
Matthews - 4401
Matthews - 8100
Matthews - 8200
Metro
Mettler Toledo 8865
Microcom 220
Monarch 1460
Monarch 9402
Monarch 9403
Monarch 9425
Monarch 9445
Monarch 9446
Monarch 9474
Monarch 9490
Monarch 9494
Monarch 9800
Monarch 9830
Monarch 9850
Monarch 9855
Monarch 9860
Monarch Renegade-4

N

Norprint - Mistral
Novexx - Fox 6300
Novexx - Fox 6320
Novexx - Fox 6500
Novexx - Fox 6530
Novexx 6000 Series
Novexx 9000 Series
Novexx Cobra
Novexx Mustang 1000
Novexx Mustang 2000
Novexx Panda

O

O'Neil LP3
O'Neil microFlash2i
O'Neil microFlash2t
O'Neil microFlash4t
O'Neil PrintPAD

P

Panda Fox - 6300
Panda Fox - 6350
Panda Fox - 6500
Panda Fox - 6550
Panduit - PTR2
Panduit - PTR2E
Panduit - PTR3
PCL - HP LaserJets & Compatibles
PNG – See Image
Pressiza 512
Pressiza 610
Pressiza 808
Pressize F-464
Printronix SL5204r
Printronix SL5304r
Printronix SLPA5204r
Printronix SLPA5304r
Printronix T2204
Printronix T3000
Printronix T3204
Printronix T3306
Printronix T3308
Printronix T3400
Printronix T4160
Printronix T4204
Printronix T5204
Printronix T5204r
Printronix T5206
Printronix T5206r
Printronix T5208
Printronix T5208r
Printronix T5304
Printronix T5304r
Printronix T5306
Printronix T5306r
Printronix T5308
Printronix T5308r

Q

Quadrel Applicator

R

RJS - TTP 142

S

Sato CL408
Sato CL408e
Sato CL412
Sato CL412e
Sato CL608
Sato CL608e
Sato CL612
Sato CL612e
Sato CT400
Sato CT410
Sato CX200DT
Sato CX200TT
Sato CX208
Sato CX212
Sato M-5900
Sato M-5900RV
Sato M-5900RVe
Sato M-8400
Sato M-8400-Long
Sato M-8400RV
Sato M-8400RVe
Sato M-8450
Sato M-8459S
Sato M-8459Se
Sato M-8460S
Sato M-8460Se
Sato M-8485S
Sato M-8485Se
Sato M-8490S
Sato M-8490Se
Sato M84Pro
Soabar - 350
Soabar - 380
Soabar - 390
Soabar - 390
Soabar - 5000
Soabar - 5500
Soabar - D-700
Soabar - PA-386
Soabar - SPX360
Soabar - SPX361
Soabar - SPX362
Soabar - Traveler
Std Register - PR720
Superior - TBL9000
Survivor - SST

T

Tharo - 112
Tharo - 112+
Tharo - Apollo 2
Tharo - Apollo 3
Tharo - Freedom
Tharo - Gemini
Tharo - Gemini 2
Tharo - Hercules 200
Tharo - Hercules 600
Tharo - Magic 300
Tharo - Orion
Tharo - Wizard
Tharo - Wizzard
Toledo 8865
TSC TTP - 244Me
TSC TTP - 248M
TSC TTP - M45

U

UBI - C407
UBI - Easycoder 401
UBI - Easycoder 501
UBI - Easycoder 501XP
UBI - Easycoder 601
UBI - Easycoder 601XP
Unimark - 400
Unimark - 400TX
Unimark - 420
Unimark - 570
Unimark - U150
Unimark - U300 Series
Unimark - U400 Series
Unimark - U500
Unimark - U501
Unimark - U522
Unimark - U522CX
Unimark - U522TX

W

Weber - 2075
Weber - 2076
Weber - 2077
Weber Legitronics 31 Series
Willet - 2300
Willet - 2600
Willet - 2600 Series 2
Willet - 2600-47
Willet - 2610
Willet - 2611
Willet - 8400
Willet - 8450
Willet - Labeljet 2600 S

Z

Zebra 105 S
Zebra 105 Se
Zebra 105 Se (152 dpi)
Zebra 105 Se (300 dpi)
Zebra 105SL (203 dpi)
Zebra 105SL (300dpi)
Zebra 110PAX3 (203 dpi)
Zebra 110PAX3 (300 dpi)
Zebra 110PAX4
Zebra 140 Xi
Zebra 140 XiII
Zebra 140 XiIII
Zebra 160 S
Zebra 170 Xi
Zebra 170 XiII
Zebra 170 XiIII
Zebra 170PAX3 (203 dpi)
Zebra 170PAX3 (300 dpi)
Zebra 220 Xi
Zebra 220 XiII
Zebra 220 XiIII
Zebra 2746e
Zebra 90 Xi
Zebra 90 XiII
Zebra 90 XiII (600 dpi)
Zebra 90 XiIII
Zebra 96XiIII
Zebra DA402
Zebra LP 2844
Zebra LP 2844-Z
Zebra PA400
Zebra PT400
Zebra QL 220
Zebra QL 220 Plus
Zebra QL 320
Zebra QL 420
Zebra QL 420
Zebra QL 420 Plus
Zebra R110PAX3
Zebra R110PAX4 (203 dpi)
Zebra R110PAX4 (300 dpi)
Zebra R110Xi (203 dpi)
Zebra R110Xi (300 dpi)
Zebra R110XiIIIPlus
Zebra R-140
Zebra R-140
Zebra R170Xi (203 dpi)
Zebra R170Xi (300 dpi)
Zebra R2844-Z
Zebra R2844-Z
Zebra R402
Zebra R4Mplus
Zebra RW 420
Zebra S4M
Zebra S600
Zebra Stripe S300
Zebra Stripe S400
Zebra Stripe S500
Zebra Stripe S600
Zebra T402
Zebra TLP 2722
Zebra TLP 2742
Zebra TLP 2844
Zebra TLP 2844-Z
Zebra TLP 3742
Zebra Z4000
Zebra Z4000 (300 dpi)
Zebra Z4M (203 dpi)
Zebra Z4M (300dpi)
Zebra Z4Mplus (203 dpi)
Zebra Z4Mplus (300dpi)
Zebra Z6000
Zebra Z6000 (300 dpi)
Zebra Z6M (203 dpi)
Zebra Z6M (300 dpi)
Zebra Z6Mplus (203 dpi)
Zebra Z6Mplus (300 dpi)

Appendix B – Symbologies

Not all Symbologies are supported by all printers. Be sure to select the correct printer for your use. All available Symbologies for that printer will be displayed in the selection box.

Bookland
Code 11
Code 16K
Code 49
Code 93
Code 128
Code 3 of 9
Discrete 2 of 5
Extended 3 of 9
Codabar
EAN-8
EAN-13
FIM
HIBC Code 3 of 9
HIBC Code 128
Industrial 2 of 5
Interleaved 2 of 5
Interleaved 2 of 5 mod 10
Interleaved 2 of 5 UPC Shipping
K-Mart non EDI
PDF 417
LOGMARS
Matrix 2 of 5
Maxicode
MSI
Plessey
POSTNET
Standard 2 of 5
Telepen
UCC/EAN-128 Random Weight
UCC/EAN Code 128
UCC/EAN Code 128 (SSCC-18)
UPC 2 digit
UPC 5 digit
UPC-A
UPC-E
UPC-D

Appendix C – Supported File Types

ASCII - Fixed Field

Each field is a specified number of characters in length. Each record ends with a new line character.

Example:

NAME	DATE	PO
FIRSTONE	010196	1458
SECONDONE	020296	947

ASCII - Delimited

Each field is separated from the next by a specific character, such as a comma.

Example: Using comma as delimiter.

Name, Date, PO

FIRSTONE, 010196, 1458

SECONDONE, 020296, 947

Appendix D – BARCODE 2000 File Type Identification

Included with this software is a complete list of all files shipped with this program. If you should incur certain error messages, you may wish to verify against that list to ensure that critical files have not been inadvertently deleted from your system.

The following identifies the role of each file type in the operation of this program.

Executable files (no extension)

Executable files are the heart of the program. They provide the command information to the computer. Any damage to these files may require reinstallation of the entire program.

.lbl

Files with the .lbl extension are label template files. These files are created in the LDS.

.ldd

Files with the .ldd extension are Data dictionary files. These files are created in the LDD, where you can set the parameters and save them. There is a Master LDD set up in BC2000 as an example.

.cfg

Files with the extension .cfg are config files. A BC2000.cfg file is delivered with the install of your software. Every time changes are made in the Configurator, this updates the BC2000.cfg file.

.pcx

The PCX file is based on a "monochrome bitmap file" and is used for graphic information, such as logos. We have included several versions of the Unibar™ logo for your practice use.

printers.typ and printcap.dat

These are critical files that allow the program to identify the specific features available with your printer type.

printers.lst

This file is created by the Configurator tool. It is the list of all the printer drivers you intend to use with the software. If you create a format and it does not print, perhaps you forgot to load the driver.

Appendix E – Creating an ASCII file with Access/Excel

ACCESS

The following is instructions on how to create an ASCII file out of Access. These examples assume you have and existing database setup, and are exporting the file for use with BARCODE 2000.

The first step to exporting a file is to determine what data fields you are going to extract. To extract certain data fields, create a query that contains the fields you want in the file. Next, select the query by highlighting the name of the query in the Queries view. After you have highlighted the query you want to export, select File then Save As/Export from the menu. A message-box pops up asking you to choose between *To an External File* or *Database*, or *Within the Current Database as*. Choose the first one, *To an External File* or *Database*, and select the OK button. The next message box called Save Query '*query name*' In… is displayed. On this screen you must change the File name: to a name other than your original query name. Then change the Save as type: to Text Files, select the EXPORT button when finished. An Export Text Wizard box will be displayed on the screen, if not see the Microsoft help on Access.

In this example, we are going to export a delimited file, so choose Delimited for a file format, and then select the NEXT button. The next screen will let you choose a delimiter; in this example we chose a Comma for a delimiter. Next set the Text Qualifier to {none}, this will take the quotes off any text. Then select the NEXT button. The next message box that appears is the final step, it displays the path and file name of the comma delimited ASCII file, so select FINISH if everything is correct.

You have just created a comma delimited ASCII file ready to be sent to the Formatter. To learn more options on how to export file out of Access, select the help in Access.

EXCEL

The following is instructions on how to create an ASCII file out of Excel. This example assumes you are exporting a file for use with BARCODE 2000. Excel will only export one sheet at a time. If you have more than one sheet to convert into ASCII format, repeat these instructions for each sheet.

After you have decided on what sheet should be put into ASCII format, choose File, Save As from the menu bar. A Save As dialog box is displayed. On this screen you must change the File name: to a name other than your original file name. Then change the Save as type: to one of the delimited file types (Tab or Comma delimited). After you have chosen the type of file to save, select the SAVE button.

You have just created a delimited ASCII file ready to be sent to the Formatter. To learn more about creating a delimited file with Excel, see Excel help.

Appendix F – Printing "label formats" to non-printers

JPEG, PNG & HTML

Most JPEG and PNG images you create with e-BARCODE 2000 will be displayed in browser for viewing and printing. The most common HTML tag you will be using to insert the JPEG and PNG file into HTML is:

<IMG SRC="filename"> (used with the JPEG 100 or PNG 100 printer)
<IMG SRC="filename" width="pixels" height="pixels"> (used with the JPEG 300 or PNG 300 printer)

NOTE – with the JPEG 300 or PNG 300 printer, your JPEG or PNG image will be 3 times the size of the original.

Example: a 4x6 inch label will be 12x18 inch when using the JPEG 300 or PNG 300. That is why you need to set the width and height of the image in the HTML <IMG> tag.

NOTE – When creating a PNG file, the file is usually smaller so from the internet it will download faster, but when actually creating the file it takes longer, because there is a longer compression period.

Default Margins

Internet Explorer and Netscape both have a default border on the top, and the left. Use the following HTML <BODY> tag to get rid of the margins in both IE, and Netscape.

<BODY marginheight=0 topmargin=0 marginwidth=0 leftmargin=0>

Printing Multiple Labels

There are different ways to print multiple labels with the JPEG printer. We recommend using "FRAMES". This will work great when you go to print multiple labels; because IE has a print-function that will allow the user to printer each frame without selecting the frame then printing.

The most basic approach to the FRAMES is this:

- Create the JPEG images
- Create a small HTML file for each JPEG or PNG image that will display that particular image.
- Create a master or final HTML that will use FRAMES for each HTML.

Example: (let's say I created 3 JPEG or PNG images with the Formatter)

Step 2 is to create a small HTML file for each image.

HTML for Image 1

```
<body marginheight=0 topmargin=0 marginwidth=0 leftmargin=0>
<IMG SRC="image1.jpg">
</body>
```

HTML for Image 2

```
<body marginheight=0 topmargin=0 marginwidth=0 leftmargin=0>
<IMG SRC="image2.jpg">
</body>
```

HTML for Image 3

```
<body marginheight=0 topmargin=0 marginwidth=0 leftmargin=0>
<IMG SRC="image1.jpg">
</body>
```

Step 3 is to create a master HTML that uses FRAMES to display each HTML file.

MASTER HTML FILE

```
<html>
<frameset rows="*,*,*">
<frame src="HTML1.htm">
<frame src="HTML2.htm">
<frame src="HTML3.htm">
</frameset>
</html>
```

NOTE -- <frameset row="*,*,*">, if there are more HTML files, then you will need to add and "*" to this tag. Example: if I had 6 HTML files that need to be displayed in FRAMES, then I would use <frameset rows="***,*,*,*,*,***">.

Appendix G – Configuration File & Print Destination Table

You can use an editor to manually edit the configuration file. This section will explain how to setup the configuration file. The BARCODE 2000 Configuration File, BC2000.cfg, contains configuration data for the site. It is stored in the "Unibar™ base directory". Normally, the Java BARCODE 2000 Configurator Module updates it. However, if the Label Formatter is running on a system without a Java runtime the user can change the file with any text editor.

The file is delivered with examples of all possible settings, many of which are commented out. The user will often be able to make changes simply by commenting out lines or un-commenting lines. *.cfg files should always be saved in text (ASCII) format.

Note: The file BC2000.cfg must be in the UNIBAR™ base directory and must have at least one valid PRT_DEST element. If not, the program will not print. The Label Designer reads the list of logical names from the BC2000.cfg file and allows the user to select a printer from that list. If the list is empty then no print destination may be selected.

General Format

The pound sign '#' is used as a comment delimiter. When this is placed on a line in BC2000.cfg, the remainder of the line is ignored.

The file is divided into sections. For example, the Printer Destination Table section, shown below describes the printer destinations configured for the site.

Each section begins with a header in square brackets. The section continues until either a new section header is encountered or the end of the file is reached.

Each section consists of a set of elements. A simple element sets one value, for example "DEFAULT=Shipping". A complex element consists of a set of simple elements. For example, the complex element PRT_DEST consists of three simple elements, as shown below in the Printer Destination Table section.

Printer Destination Table

The output of the BARCODE 2000 programs (Designer, Formatter …) is a print stream that can be sent to one of three types of destinations:

- Sent directly to a device, (e.g., "/dev/lp0")
- Piped to a program (e.g., "lp -d LabelPrinter)

The user can define a number of printer destinations in BC2000.cfg. Each entry consists of a logical name, a physical name and a destination type. This provides a way to use a simple logical name to refer to a sometimes-complex physical destination (such as "lp -d LabelPrinter -m -q 3 -s ").

This also provides a way to change physical printer destinations without changing the application program. Simply update the BC2000.cfg file and the applications and operators continue to use the logical names they used before.

To print to a particular printer destination, the user specifies the logical name of that printer. If no logical printer is specified, the program uses the default as specified in the BC2000.cfg file.

An Example of the Printer Destination Table:

```
[Print Destination Table]

PRT_DEST =
{
      Logical = Shipping
      Physical = lp -d LabelPrinter
      Type = PIPE
}

PRT_DEST =
{
      Logical = Printer3
      Physical = \\Network\HPLASER
      Type = DEVICE
}
PRT_DEST =
{
      Logical = Receiving
      Physical = lp -d LabelPrinter
      Type = PIPE
}
DEFAULT = Shipping

[File Type]
Type = DELIMITED
#Type=FIXED
DELIMITER =~
#DELIMITER =,
#DELIMITER =.
TRIM_SPACES = Y
#TRIM_SPACES = N
TRIM_QUOTES = Y
#TRIM_QUOTES = N

[Logging]
LOG_FILE = STD_OUT
LOG_LEVEL = ERRORS_ONLY

 [Capacity]
MAX_LABELS=16
MAX_RECORD=2048
```

The string, “[Printer Destination Table],” is the section header. The PRT_DEST element defines a printer destination.

Logical
"Logical" is a name assigned by the user and is the name by which users refer to the printer. It can be any useful name, such as ACCOUNTING or DOCK_10. It is a text string of up to 40 characters. It cannot contain any command line delimiters (space, comma, equal sign, slash, etc.) because it may be used on the command line.

The number of print destinations, and the number of Logical names, is not limited by BARCODE 2000. Several Logical names can refer to the same physical name. This can be quite useful in shielding applications and operators from the details of printer management. Logical names should not be duplicated. If they are, the first definition will be used and any subsequent definitions will be ignored.

Physical
The meaning of the physical name depends on the destination type. For DEVICE it is the device name. For PIPE destinations, 'physical name' is the shell command line that is passed to the system when opening the pipe stream. The command is executed and a pipe is opened from the Formatter to the program started with the command line.

The number of physical devices is limited. The program will only write to the number of physical devices for which the user is licensed. This is implemented by counting from the top of the Printer Destination Table section; so, if a user is licensed for five printers, the program will write to the first five physical print destinations listed in the configuration file.

Type
Type may consist of one of two keywords {DEVICE, PIPE}.

- DEVICE - is used for writing directly to I/O devices such as /dev/lp0, /dev/tty1a, etc.
- PIPE - is used to specify a pipe command string.

Default Printer
The DEFAULT element specifies the default printer by its logical name. If no default is specified, the first entry is taken as the default

File Type
This section of BC2000.cfg defines a default file type and delimiter for input to the Formatter. The default built into the Formatter is Delimited file type with a delimiter character of '~'. The defaults can be overridden on the command line or in a job file.

Defining a Delimiter
This is where you would change the delimiter for input to the Formatter. Please see the previous item.

Capacity

This section, in general, the larger the settings the more memory consumed. Memory for many of these items is allocated up front, for the sake of simplicity; so it is allocated even if it is not used. Thus, if memory usage is a concern, you may want to keep these settings on the lean side.

Max Labels

The number of different label templates that will be loaded in a printer memory at one time. If this number is 5, on the 6th distinct label type encountered during a single printing, the constant part of the labels and graphics will be deleted from the printer memory. New labels and graphics will be loaded until the max number is reached again.

Max Record (Length)

This setting defines the maximum length of a data record being sent to the Label Formatter. Records longer than this will be truncated. Truncated data may cause incomplete labels.